Vrije Universiteit Brussel

FACULTY OF ENGINEERING
Department of Electronics and Informatics

# Video-based Tracking of Physical Documents on a Desk

Graduation thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Applied Sciences and Engineering: Applied Computer Science

Sone Nsime Ngole

Promoter: Prof. Dr. Beat Signer
Advisor: Dr. Bruno Dumas


JANUARY 2014

Vrije Universiteit Brussel

FACULTEIT INGENIEURSWETENSCHAPPEN
Vakgroep Elektronica en Informatica

# Video-based Tracking of Physical Documents on a Desk

Afstudeer eindwerk ingediend in gedeeltelijke vervulling van de eisen voor het behalen van de graad
Master of Science in de Ingenieurswetenschappen: Toegepaste Computerwetenschappen

Sone Nsime Ngole

Promoter: Prof. Dr. Beat Signer
Advisors: Dr. Bruno Dumas


JANUARI 2014

# Abstract

Currently, physical and digital documents tend to stay in their world, without any direct relationship linking them. However, a lot of physical documents are printed from digital documents and conversely, digital documents can be scanned versions of printed papers. Furthermore, the organization of piles of physical documents on a desk hints at shared semantic features between a set of documents. This thesis explores an approach to link or re-link physical documents with their digital counterpart. This integration will be done by designing a system that uses an overhead digital camera to recognize, identify, localize and track paper documents on the physical desk space in real time (or offline by means of a pre-recorded video stream) and automatically matching them against an image database of electronic documents. The system locates each paper document that is present on the desk and reconstructs a complete configuration of documents on the desk at each instant in time. Also, the system tracks paper documents in a pile by computing the difference between consecutive video frames from the overhead camera to detect if a document has been added to a pile of documents or removed from a pile of documents and automatically updates the digital model of the desk. The system recognizes, identifies and tracks paper documents on the desk by computing the descriptive local features of the video frames, based on histograms of edge orientation in a window around each point in the video frame, and matches these computed local features against pre-computed local features of an image database of electronic documents. The Speed-Up Robust Feature algorithm was employed in the thesis for the computation of the descriptive local features of the video frames and the image database of electronic documents.

# Contents

# 1 Introduction

## 1.1 Problem Statement

The human memory, which is used for the organization and management of information by creating associations between different information items [90], has some limitations. The most profound limitation of human memory is the storage capacity and the loss of associations needed for recall [89]. Due to human memory limitations, paper and electronic technologies (such as personal computers, flash drives, tablets, smart phones etc.) are preferred for the storage of information. However, paper documents are easy to manipulate and annotate but not suitable for storage and retrieval [46] while digital documents, which are not well suited for annotation, are best suited for storage and retrieval [98].

Various techniques exist to manage information in the physical world such as piling and filing [89]. On the other hand, there exist information management software such as computer operating systems, databases etc., aimed at managing digital information in an electronic environment. However, for a better management of information in both the digital and physical world, there is a need for the integration of the physical and digital worlds in a way that utilizes the benefits of both worlds. Researchers have been working on the integration of the physical and digital workspaces leading to many trends including the desktop metaphor, computer augmented environments, electronic tags, interactive papers and the digital desks.

## 1.2 Research Objectives

Despite the efforts of researchers [98] [26] [46] [88] over the last three decades to design a system which manages information in both the physical and digital world, there is still a lot of work to be done in this domain. The global focus of this thesis is the management and organization of information in an office environment, specifically on the office desk work space. The office desk contains both physical information such as paper documents, sticker notes, books etc. and digital information stored in a digital environment such as computers. Since most of the physical documents on the desk have their digital counterparts in the digital environment, designing a system that provides a link between them will be a profound step in the management of information on the office desk. Various technologies exist for the identification, and to a greater extent the tracking of paper document, such as RFID tags and barcodes, but these technologies require the physical placement of electronic

tags on every paper document on the desk and some require specialized readers for identification. These specialized readers and electronic tags limit the interaction between the user and the paper documents. However to overcome these limitations and enable the users to interact with the paper documents naturally, a vision-based technique is required. With a vision-based technique, there is no need for the placement of any physical artifact directly on the paper documents for identification. All that is needed is the video processing of the activities on the desk. The specific focus of this thesis is creating a digital representation of the organization of the paper documents on the desk by recognizing, identifying, and tracking the paper documents on the desk by matching descriptive local feature of video frames from an overhead camera against an image database of electronic documents.

## 1.3 Thesis Structure

The next chapter takes a look at the paper document organization techniques in the physical space, such as piling, filing, mixing and what organization technique is used most frequently by people in the organization of paper documents on the desk. Also a brief discussion on paper and digital documents will be done, focusing on their advantages and disadvantages. The DigitalDesk [98] is introduced as one of the first solution to design a PIM that manages information in the physical and digital world by integrating paper document and digital documents. Also, some ergonomic studies are reviewed for the design of a digital desk. To conclude the chapter, a review of related works on the digital desk will be discussed focusing on their limitations on tracking paper documents in pile.

In chapter 3, a comprehensive review of vision-based document processing methods will be done. The chapter concludes with a brief look at Optical Character recognition, which will be one of the approaches used in the thesis to recognize, identify and track paper documents in a pile.

Chapter 4 presents the tools and frameworks used for the design and implementation of a digital desk system. Also the technical issues and difficulties encountered in implementation will be discussed and conclude this thesis.

Chapter 5 presents the future works to be done to improve the system.

# 2 State Of The Art

## 2.1 Personal Information Management

People use information daily in their workplace and home. In the office environment, information is mostly stored either physically as paper documents, books, notepads, etc. or electronically as digital documents in computers, flash drives, tablets, mobile phones, etc. As information gets more cumbersome, the difficulty to find the right information satisfying a specific information need arises. This difficulty is caused by the fact that the human memory can only handle a fixed amount of information at a time. The difficulty to find information gets even worse with information stored physically. Since the human memory can only hold a limited amount of information at a time, organizing information is of great importance for the ease of access and retrieval of information. Personal information can be managed and organized in both the digital and physical spaces. However, since this thesis is focused on the management and organization of paper documents on the physical desk, we will concentrate on the approaches on information management and organization in the physical space.

Thomas Malone, in his publication from 1983 [58], defines two strategies of information organization on the physical space, which includes *filing* and *piling*. Files are units consisting of individual elements, which are explicitly titled and arranged in some systematic order. These individual elements as described by Malone [58] are seen as information carriers (e.g. paper), larger objects which are composed of elements (e.g. folder) and groups which group individual elements (e.g. grouped folders). Furthermore, these elements can be explicitly titled and systematically ordered in a user-specific way, mostly alphabetically or chronologically. An example of these concepts can be seen in figure 1.

Also, grouped elements can be seen as a file as long as the whole group is ordered in a systematic order and the group is titled. An example of a filing organization is shown in figure 2.

The filing organization strategy had already been in use in the mid-1800 where people filed information items, mostly books, in filing systems with explicit labeling and order. This behaviour of organizing personal information items led to the well used *Universal Decimal Classification Model* which is a model where categories are labeled by the decimals and the order is fixed [59]. Nevertheless, in the personal information space, it is not always desired to be restricted to the filing requirements of labeling and ordering of information elements. This leads more in parallel to the start of piling

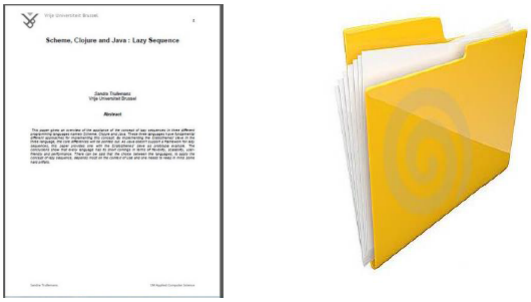


Figure 1: *Elements by Malone [58]. Left: Paper as an element, Right: Folder as an element*

information elements by people. With piles, the individual elements are not necessarily titled and they are not, in general, arranged in any particular order. Elements forming part of the pile can be titled but the entire pile cannot. The dynamics of pile creation often give the piles a haphazard order, but this is not usually an intentional arrangement. The spatial location of piles is important in the retrieval of the individual elements they consist of. One of the reasons that lead to piling is the fact that people have difficulties to decide on a classification structure (e.g. alpha-label, contextual or chronological) which will ease the process of retrieval. Another reason for the use of pile organization strategy is the difficulty to label an item with enough information about the item itself but that still fits in the context of the classification structure.

Each pile on a desk mostly contains information about different concepts or different parts of a single task to be done by the user. Piles on a desk are not always separated as can be seen in figure 3.

Elements of different piles can overlap with each other making the physical desk organization even more haphazard. This fuzziness comes from the classification problem [58] where some items could belong to more than one concept. A compensating strategy to minimize the fuzziness is by ordering the piles in such a way that the papers belonging to two related piles are just sliding somewhat out of one pile pointing to the other related pile. A summary of the properties of the filing and piling organization strategies can be seen in figure 4.

The piling and filing organization strategies do not cover all organization behaviour in the physical space. An element can be neither a file nor


Figure 2: *An illustration of a filing organization*

a pile (e.g. a titled folder which contains unordered papers) as shown in figure 5. This organization strategy is mostly used in our daily information organization.

In addition to filing and piling, there exists a third strategy of information organization known as *mixing* [89]. The mixing strategy contains a mixture of both titled and untitled elements and may be explicitly ordered. An example of mixing is the use of labeled ring binders containing semi-ordered publications. The mixing organization strategy is also used by people in the physical information space along side piling and filing. A detailed study on the degree of use of the piling filing and mixing organizational strategies on the physical desk space was done by Trullemans [89] and an illustration of their usage from 'never' to 'a high degree' in use, is shown in figure 6.

To better manage information, either physically stored or digitally stored, there is a need for Personal Information Management (PIM). PIM is the storage, organization, and retrieval of information by an individual for his/her own use [6]. In an office scenario, a better PIM leads to increase in productivity. Ideally, the perfect PIM will enable people access the right information at the right time, in the right place and in the right form. Also, the perfect PIM would enable users to manipulate, organize, annotate, group and link information to accomplish their goals. As stated earlier, information can be stored physically as well as electronically. This means that techniques

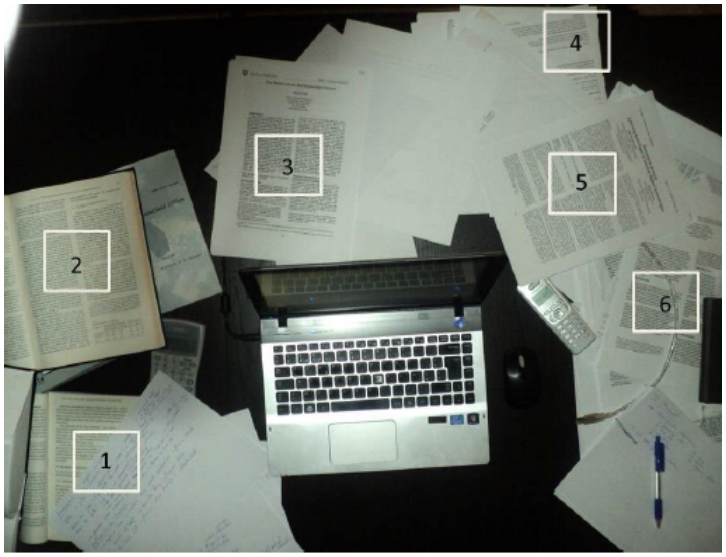


Figure 3: *spacial advantage of piles*

for managing physical information, to a greater extend, are not well suited for the management of information in the digital world. Bridging the gap between the physical and digital world will eliminate the categorization of information as being physical or digital which will lead to a single PIM for information across both worlds. This thesis is focused on the management and organization of physical paper documents and their digital counter parts on the desk by means of integrating the physical and digital worlds.

## 2.2 Paper and Digital Documents

The physical desk deals with physical papers and papers have properties that people just cannot seem to give up, making it resilient [98]. The physical paper has the properties of being three dimensional, portable, high resolution, tactile, easy to manipulate and annotate [98]. On the other hand, digital papers are inherently suitable for computational operations such as storage, keyword search, retrieval, sharing and version management [46]. However, the major difference between the physical paper and electronic document lies in the task of reading [67]. A fairly substantial body of literature comparing the reading of physical paper versus digital paper can be found in the psychological, human factors, and ergonomics literature (see [21] [20] [36] for comprehensive reviews). The majority of studies focus on outcome measures

| Files | Piles |
|---|---|
| Users collect more information but access less it less frequently | Users collect less information but access it more frequently |
| Support long-term storage | Support for short-term storage |
| Loss of context in classification | Preservation of context |
| Small amount of spatial references | Large amount of spatial references |
| Location and label used in the retrieval process | More emphasis on location in the retrieval process |
| Need a lot physical space | Restricted use of the physical space |
| No reminding function | Good reminding function |
| Stable in time | Less functionality supported when increasing the amount of piles |

Figure 4: *Properties of piles and files from [89]*

of reading, such as speed [61] [100], proof-reading accuracy [19] [30] [100], and comprehension. A lesser effort has been devoted to looking at process differences between reading on paper and reading on screen such as how readers look at text in terms of eye movements [29], how they manipulate it [73], and how they navigate through it [21]. Some major differences were witnessed on the reading task between paper documents and digital documents by the research of Kenton OHara and Abigail Sellen [67]. These differences in reading tasks between the paper documents and digital documents were grouped into two categories namely: annotation and movements.

1. **Annotation While Reading:** The ability to annotate while reading is important in enforcing an understanding of the source document and helps in planning for writing. There are three major differences between physical paper and digital paper regarding annotation while reading:

   (a) Annotation on physical paper was relatively effortless and smoothly integrated with reading compared to digital paper annotation which was cumbersome and detracted from the reading task.

   (b) Physical paper supports annotation of the source document itself while digital paper conditions do not provide enough flexibility to

Figure 5: *neither file nor pile*

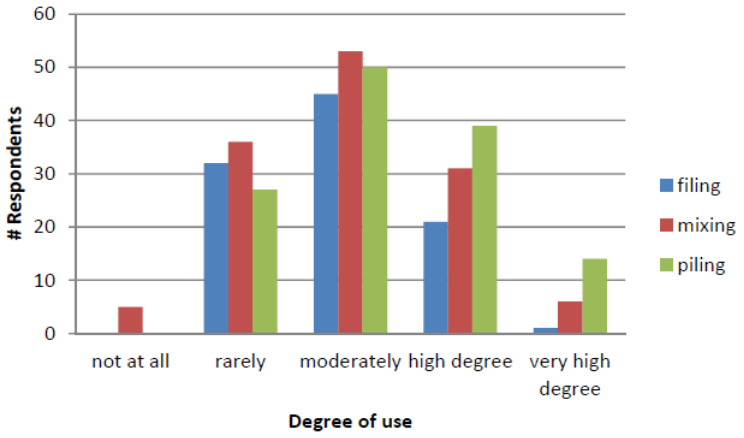


Figure 6: *General degree of use for piling, filing and mixing from [89]*

do this not does it support the richness and variation of annotating on physical paper.

(c) With paper document, note taking can be done quite frequently and interleaved with the reading process which makes the combined process of reading and annotation very smooth. However, with digital documents, reading is interspersed with long periods of editing, or note taking is done after reading while reading with little references back to the source document.

2. **Movements Within and Between Documents:** Movements through documents is important for information organization, reference and for checking understanding. There are four major differences between

physical paper and digital paper regarding movements within documents.

(a) With physical paper, navigation through paper was quick, automatic, and interwoven with reading. On the other hand, with digital paper, moving through papers is slow, laborious and distract from reading.

(b) Physical paper gives the ability to perform two handed movements which allow readers to interleave and overlap navigation with other activities, and allow temporary commitment to interim activities. Whereas, with digital paper, movements require breaking away from ongoing activity and committing to navigation activities because it is one handed, not always accompanied by immediate feedback and spatially constrained to active areas on the screen.

(c) The tactile properties of physical papers enable it to support navigation and to implicitly access document length. On the other hand, the digital paper fails to make use of explicit cues such as page length to access to document length.

(d) The fixity of information with respect to physical paper pages supported incidental memory for where things were, which in turn supported search and re reading activities. The inability to see a complete page may undermine use of this feature with digital papers, but it appears pictures were used as anchor points.

The physical and electronic worlds might often be related but they are very separate regarding interaction styles functionality [98]. We use our fingers, arms, 3D vision, ears, kinesthetic memory as tools to manipulate physical paper on the physical desk. Using the above mentioned tools to manipulate physical paper is easy because we have developed the natural skills that are embedded deeply in our minds and bodies to work with these natural tools. Manipulating digital papers does not take advantage of these natural skills. There is also a difference in functionality between the physical desk paper and its electronic counterpart. Unlike the electronic paper, the physical paper has no direct access to database queries and spreadsheet calculations available, but it has portability, tangibility and universal acceptance.

## 2.3 Integrating Paper and Digital Documents

Over the last few years, there has been a significant growth in the amount of research dealing with the integration of the physical and digital worlds [93] [14] [101] [37] [82].

Electronic printers and scanners provide a link between the electronic and physical world by acting as a gateway across both worlds. However, these devices are not very interactive and the process of moving entire documents in or out of electronic desktops is very inconvenient. The following is a brief description of some attempts to integrating paper and digital documents.

1. **The Desktop Metaphor:** The classic approach to merging the electronic and physical desk is the desktop metaphor which allows the use of direct manipulation of virtual objects on the digital desktop [19]. The desktop metaphor ensures that users take advantage of their knowledge of the physical world by making the electronic workstation analogous to the physical desk.

2. **Computer Augmented Environments:** This is an approach aimed at fusing the electronic and physical worlds by using computers to augment objects in the real world. This can be done by making an environment sensitive with infra-red, optical sound, video, heat motion, light detectors, etc. Computer Augmented Environments merge electronic systems into the physical world instead of replacing it. Example works in computer augmented environments include the work by Knowlton [47] which was a system that combined the flexibility of a computer generated display with the tactile and kinesthetic feel of physical buttons. Others include VideoDraw [87],VIDEOPLACE [50] and the Mandella system [92].

3. **Electronic Tags:** Electronic tags have been used as a way to bridge the gap between the physical and the digital world [93] [27]. Various interesting methods exist for tagging physical paper such as bar codes or glyphs [40]. The main approach to bridging the physical and digital worlds via electronic tags is to take everyday objects which already have some useful purpose independent of any electronic system and to augment those objects through embedded RFID tags [93].

4. **Interactive Paper:** Interactive paper is an interesting means to fuse the physical and digital worlds. This method was emphasized by Mark Weiser [96] where he described a scenario of how intelligent paper might be integrated into future working environments. Two basic methods exist for integrating physical paper with digital information, namely the *electronic paper* approach which aims to make existing devices as

paper-like as possible and the *interactive paper* approach which focuses on augmenting regular physical paper by linking it to supplemental digital information and services [82]. The current interactive paper solutions to integrate the physical and digital worlds are based on the Anoto's digital pen and paper technology [82], where a camera is integrated in a digital pen. The pen reads a unique printed dot pattern on a physical paper and hence can detect the pen's position within the given physical paper. There have been many realized interactive paper solutions based on the digital pens working namely; butterflyNet [104], the Paper Augmented Digital Documents [35], and PapaerProof [95].

## 2.4 The DigitalDesk

The concept of the Digital Desk, introduced by Pierre Wellner [98], was designed to bridge the gap between the paper documents and the digital documents, creating a hybrid work space that encompasses the advantages of both physical and digital world interactions. Wellner's design, called DigitalDesk [98], consisted of a video camera mounted above a desk, a computer driven projector mounted above the desk as well, and a LED-tipped pen. The camera points downward onto the desk and its output is fed through a system that can detect where the user is pointing (using the LED-tipped pen). The projector enables electronic objects to be projected on the physical desk as well as all physical objects on the desk which mostly paper documents. The set up is shown in figure 7.

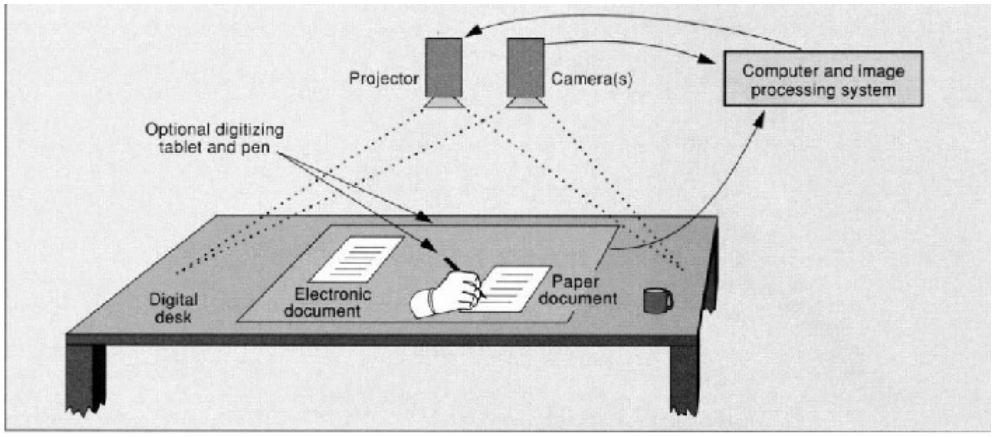


Figure 7: *DigitalDesk setup*

The DigitalDesk as designed by Wellner [98], has three important characteristics namely, projection of images onto the desk, responding to interaction with pens or fingers and recognizing documents placed on desk.

Wellner's DigitalDesk [98] was implemented with various applications and also had some implementation issues. A brief description of these applications and implementation issues is outlined in the following section.

### 2.4.1 DigitalDesk Applications

A range of applications are made possible by the DigitalDesk designed by Wellner [98]. Some working prototypes were implemented to varying levels of robustness. A few of these prototype applications are described below.

1. **Calculator:** The projector mounted above the desk projects an interactive electronic calculator interface. The Digital Desk calculator provides an alternative means of entering numbers. It allows people to place ordinary paper documents on the desk and simply point with a pen or bare finger and a rectangle is projected in front of the finger to indicate which number is selected. When the user taps on the desk, the system reads this number with a camera, recognizes the digits, and treats them as though they had been typed into the calculator [98]. the results obtained from the calculator can be projected back onto the paper. The system uses image differencing to follow the finger or pen and it detects taps by listening with a microphone attached under the desk. The field of view of the high resolution camera covers the portion of the physical paper where the user is currently interacting with either fingers or pen. This field of view is used for Optical Character Recognition (OCR), while the camera used for finger tracking covers the entire desk surface.

2. **Desktop Translation:** This application was implemented by Wellner and Newma [64] in which French documents can be read at a desk in their paper form and the user can simply point at unknown keywords. The system extracts the root of words, looks them up in a French to English dictionary and displays the definitions in an electronic window projected unto the desk, allowing the user to point to the location where the translation should be placed on the desk. This system uses the low resolution image it gets from the overhead camera to recognize which of the pre-screened images it corresponds to. It uses the shape

of text margins, gaps between paragraphs and gaps between words to recognize documents in a resolution-independent way. This means the camera does not need to be zoomed in close to the desk, and the single wide-angle view of the desk can be sufficient to access the finest details of recognized pre-scanned documents. The desktop translator can be seen in figure 8.

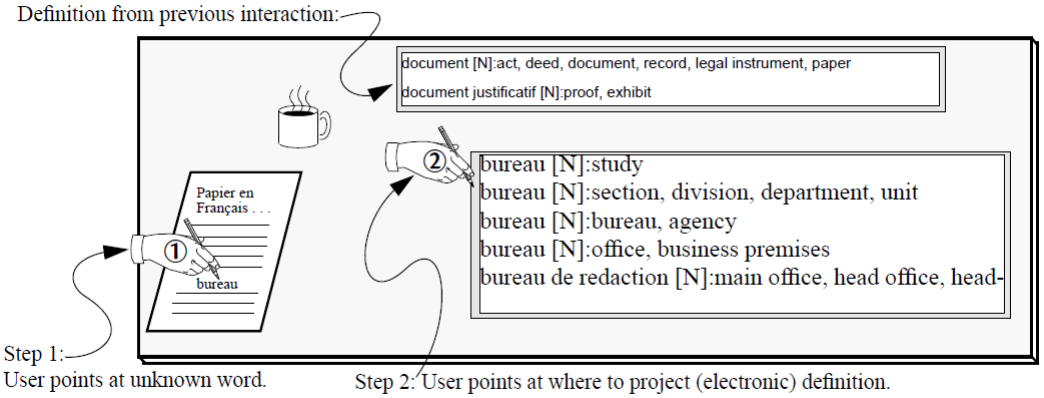


Figure 8: *Translation On The DigitalDesk*

3. **PaperPaint:** This application enables the user to perform the action of copy and paste between the digital and the physical world. This application allows users to construct a mixed paper and digital drawing. A sketch on paper can be digitally selected by sweeping out an area of the paper with a stylus; the projector displays a rectangle on the paper to indicate what is selected. When the stylus is raised, the system snaps a picture, and the projected rectangle is replaced by a thresholded electronic copy of the area. This copy can then be moved about and copied to other parts of the paper. Sliding this electronic copy over the drawing to place it somewhere else is very similar to sliding a paper copy.

4. **DoubleDigitalDesk:** One of the constraints of the physical paper document is the aspect of sharing. Two people on different continents cannot manipulate the same physical document. This is overcome by this DigitalDesk application. The DoubleDigital desk makes it possible to share and manipulate real paper documents. In this application, each DigitalDesk continuously grabs images from its local desk and

projects scaled, thresholded images from the remote desk. The result is that both users see what is on both desks.

### 2.4.2 Implementation Challenges For The DigitalDesk

The digital desk as described by Wellner [98] had several implementation issues. Some of these issues are discussed below.

1. **Interaction on the Desk:** Challenges involving user interaction on the desk can be split into two domains: input and output challenges for interaction with desk. The input challenges consist of video-based finger tracking, object selection and automatic camera calibration while the output challenges involve the projector displays.

   **Input Challenges:**

   (a) **Video-Based Finger Tracking :** One way of user interaction on the DigitalDesk with bare fingers is through video-based finger tracking. This means obscuration of fingers can be a huge problem in many situations and applications but since the hands have a limited range of motion and they mostly remain in a 2D plane, when interacting with the desk, this is not much of a problem. Pointing to smaller objects like single words or letters could also be a challenge because the bare finger is too thick and causes occlusions in most of the cases. This challenge could be roughly solved by also providing a means to use smaller objects such as pens as pointing devices for desk interactions. video based finger tracking can be challenging, but there are a wide range of interaction techniques possible using video-based finger tracking [49]. One of such techniques described by Krueger [49] relies on hands being viewed against a plain background in order to make them easier to distinguish. Unfortunately this technique cannot be used in the digital desk described by Wellner [98] because his model took into consideration not only documents on the desk, but other physical objects like pens, coffee cups, etc, which would make it difficult to distinguish these objects with the finger via video processing. Another technique is looking for motions, since the finger is the only consistently moving object on the desk. This is done by examining successive image frames of the desk and trace the path of the moving object (finger in this case). This can be shown by the illuminated traces in figure 9.

Figure 9: *Examining Successive Image Frame for Finger Tracking*

(b) **Object Selection:** One method for the user to select an electronic object is by tapping on the desk. This cannot be determined solely by the overhead mounted camera. One solution to determine this is to attach a microphone to the bottom of the desk have and the system monitor the amplitude of the signal to determine when the user taps on the desk. This is challenging since the system can confuse legitimate taps for object selection with external noise like bumps on the desk, hand claps and other similar sounding noise. There must be a perfect synchronization between the finger-following and tap-monitoring tasks because just a little lag of the image processing system will report the finger in the wrong place at the time of the tap. Another way to detect tapping is to use a touch screen. Unlike the microphone, it can provide dragging information as well as extra location data. A problem with desk-based touch screens is that users tend to rest their hands on them and everything touched can be interpreted as input.

(c) **Automatic Camera Calibration:** The DigitalDesk [98] requires calibration to support interaction on the desk, projected feedback, and selective grabbing. The positions on the display must be mapped to corresponding positions in the frame grabber in order to support grabbing of selected areas on the desk.

Optical distortions such as keystoning and vibrations caused by air conditioners or slamming doors causing movements are a few factors that conspire to make self calibration challenging. Another challenge is obtaining data for calibrating the camera to display. There are various methods used to get data to calibrate the pointing device and the camera to the display. For the pointing device, a series of points are displayed and the user is prompted to touch them with a pointer. Obtaining data to calibrate a camera can be done using several methods. A good and accurate mapping approach is to project an image that can be located by the image-processing system, allowing the system to self-calibrate without any assistance from the user [98]. Another way is using a four-point calibration system to compensate for rotation and keystoning. To calculate the mapping from four points it uses the equation in figure 10. With four point pairs, the two sets of four simultaneous linear equations can be quickly solved by Gaussian Elimination to find the values of Cl-8. Then, a fifth plus mark is projected and its location is checked to make sure it is close enough to the position predicted by the mapping.

**Output Challenges :**

(a) **Projected Display:** The projector mounted above the desk projects images on the desk. This computer generated image can be superimposed onto paper documents which is necessary for creating merged paper and electronic documents. As is the case for most digital displays, there has to be a size versus resolution trade-off. However, one of the main issues with overhead projection is obscuration. The user's shadows can obstruct the digitally projected images on the surface of the desk. Also light intensity of the room could also be a problem as bright sunlight in the room could make the projected images unreadable.

2. **Reading Paper documents:** Reading paper documents on the desk is the major part of the DigitalDesk [98] and it is still an ongoing research. The digital desk designed by Wellner [98] subdivided reading paper documents into separate tasks, namely: image capture, thresholding and character recognition. Image capture is done by the camera mounted above the desk and these images are fed to an image processing system whereas thresholding is done on the images for character recognition (details on thresholding and character recognition will be discussed in subsequent chapters). The focuses of Welner's [98] design

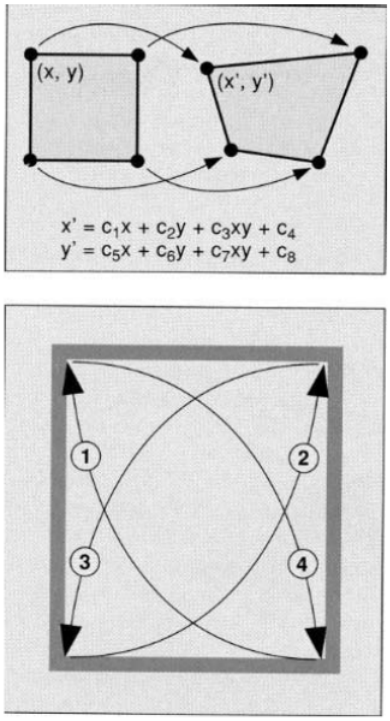


Figure 10: *Four Point Warping*

of the DigitalDesk and subsequent researches have been mostly based on the seamless transition from the physical and digital objects. This focuses more on the physical manipulation of the electronic objects projected on the physical papers and less research has been based on paper recognition and identification.

All of the above mentioned challenges are strictly related to Wellner's DigitalDesk design [98]. For the design of a generic cross-media desk with a seamless transition between the physical and digital worlds, some ergonomic challenges have to be taken into account. The following sections give a brief description of some of these ergonomic challenges.

## 2.5 Ergonomic Studies for Generic Digital Desk Design

**Visual Ergonomics:** Visual ergonomics deals with factors that determine how well contents can be viewed on a screen by users (mostly for reading tasks). Several studies have been carried out to explore these factors and the

following factors have been found to be of great importance.

1. **Display Properties:** Display properties play an important role for visual performance fatigue of the user. The two most important display properties are resolution and contrast. Studies performed on reading performances for different screen resolutions show that display resolutions lower than 120 ppi have some adverse effects on reading tasks compared to paper [66]. So, it can be suggested that displays need a resolution equivalent to printed text (300 dpi) in order to achieve reading performance comparable to paper.

2. **Perpendicular View:** In touch screens, there is always a constant offset between intended touch positions and absolute touch positions when a user's line of sight is not perpendicular to the screen. This can affect user's performance in visual search tasks. The most convenient monitor setup for visual search task is one which is curved around the user.

3. **Monitor Placement:** Monitor placement is also an important property for visual and musculoskeletal strain. Studies strongly suggest that a line of sight of about 9-10 degrees below horizontal offers the best trade off between visual and musculoskeletal strain and users prefer viewing distance of between 55 and 60 cm [66].

4. **Viewing Distance:** Studies suggests a viewing distance of at least 60 cm is adequate to reduce visual fatigue [20]. At a viewing distance of of 60 cm and assuming an angular resolution of 0.02 degrees for the human eye, a display resolution of 120 ppi would be sufficient. For a viewing distance of 30 cm a display resolution of 240 ppi would be needed.

   **Touch Ergonomics:** Much research has been done on the tasks best suitable for pointing devices and direct touch. Each of these interaction methods have their advantages and disadvantages depending on the task. Direct-touch interaction is well suited for bi-manual input and it is superior to other input methods when tasks involve relatively large targets. Several factors affect the performance of the user in direct touch tasks.

5. **Size:** The size of the interactive surface greatly affects different tasks performances especially sorting tasks. Studies show that a desktop sized surface is too large for sorting tasks as screen contents were

placed in the peripheral viewing area and tablet sized touchscreens were deemed too small by most users' [25].

6. **Placement:** The placement of the display can also affect the performance of some tasks. Most users tilt their display towards them if the display is placed horizontally. Users find horizontal displays to be in their way taking desktop and affecting their task performances.

7. **Angle:** Reports on studies show that sitting users prefer interacting with touchscreens tilted 30 degrees towards them from the horizontal as being the least fatiguing.

## 2.6 Works Inspired by the DigitalDesk

Over the last two decades, there has been numerous researches based on the digital desk concept. This has led to a number of alternate DigitalDesk [98] designs [99] [46] [88] [86]. However these varying designs have a common number of design factors that have to be taken into consideration including user interaction with the desk [98] and user annotation on the digital objects [68]. Each difference in the implementation of these factors creates a new variation of Wellner's DigitalDesk [98]. These variations span from models similar to the original design by Wellner [98], which involved a desk, projector and monitor with interaction made possible by gesture recognition via finger tracking to more complex models involving huge touch screens or surfaces [99]. Some designs try to eliminate the idea of separate desk (horizontal) surfaces and the monitor (surfaces) by blending these surfaces into one seamless interactive screen [99].

Each model variation emphasizes on different aspects of the augmentation. The following subsections describe in detail some of these variations of the DigitalDesk [98].

### 2.6.1 Blending the Vertical and Horizontal Surfaces

The vertical and horizontal surfaces can be used as interactive displays in the design of a desk which aims at bridging the gap between the physical and digital worlds. The vertical and horizontal interactive displays expose specific assets and drawbacks [97], and the choice of the appropriate angle depends on the user's tasks. Vertical displays are the established output technology in desktop environments where keyboard and mouse provide efficient input devices. The horizontal displays are more efficient for drawing tasks, such as annotation and graphic design. The idea of modeling a digital

desk which blends the horizontal and physical surfaces was shown in Tognazzini's Starfire concept video from 1994 which offers a horizontal and a vertical interactive surface [88]. The central part of the Starfire is a digital desk incorporating a large vertical display which is curved around the user. The horizontal desk is mostly used for reading and annotating tasks while the vertical part is used for video conferences. This idea was later on implemented by Microsoft in 2007 with the presentation of the DigiDesk. This consisted of a slightly tilted MS Surface with an additional vertical display along its longer side. In 2009, the BendDesk was also presented by Weiss et al [97] which was a digital desk concept that combined a horizontal and vertical interactive surface, connecting them with a curved segment. With the design of the BendDesk [97] a user cannot look over the top edge of the desk and the absolute vertical surface makes direct-touch interaction fatiguing. These were major drawbacks for the design which which was later on rectified by the design of Wimmer et al [99] in the "Curve". The "Curve" [99] is a digital desk concept that blends a horizontal and a vertical interactive surface. The Curve concept tries to bridge the gap between the physical desktop and the computer screen by blending both into one large interactive surface. To design well a desk which blends the vertical and horizontal surfaces into one seamless interactive surface some ergonomic factors need to be taken into consideration. Wemmer et al. [99] established some guidelines based on some visual and touch ergonomics, in the design of their digital desk The Curve [99]. These guidelines are summarized below.

1. **Provide Ample Resolution:** A physical resolution of at least 120 to 240 ppi should be offered for reading tasks.

2. **Maximize Screen Real Estate:** Users prefer large interactive surfaces for laying out multiple objects spatially. Therefore a digital desktop should be at least as wide as a user can reach with their hands.

3. **Allow Direct Touch Interaction Across the Whole Display:** Direct interactions are preferred and are faster than mouse input for many selection tasks on a touch surface.

4. **Offer Both Horizontal and Vertical Surfaces:** The preferred planar position of the surface (horizontal or vertical) depends on the task at hand. A nearly vertical display is suited for reading tasks while a horizontal surface is preferred for annotating and navigating digital and physical documents.

5. **Support Dual Use:** A digital desk should offer the same dual use advantages of the classic wooden desk where user are able to place other objects like books, papers, personal gadgets, coffee cups, pizza, etc.

6. **Support Alternative Input Devices:** Interactive surfaces should also be able to support alternate input devices especially for digital desks where the user might navigate a directory tree, drag a document towards himself and many more. Therefore digital desks should support a multitude of input devices that offer ergonomic advantages for certain common tasks. Keyboard, mouse, pen, multi-touch are essential.

7. **Reduce Visual and Musculoskeletal Strain:** A digital desk should generally conform to established ergonomic guidelines. The line of sight should be perpendicular to the display [7] and be inclined at about 10 degrees downwards from the horizontal [71].

8. **Allow Users to Adjust Parameters:** There should be as many adjustable physical parameters of the interactive surface as possible since the ergonomic requirements between users vary greatly.

**The BendDesk :** BendDesk [97] inspired by the Sun Starfire vision video [88] is a hybrid desk environment that blends the horizontal and vertical surfaces into one piece which is seamlessly merged into a curve. This seamless merge helps to overcome the issue where users tend to perceive the vertical and horizontal surfaces as isolated areas. BendDesk allows to perform dragging gestures from one display to the other and supports direct touch and pen input areas. The BendDesk consists of a bended acrylic board that divides the surface into three interactive areas. A 39in X 15in horizontal tabletop is mounted at a height of 28in to allow comfortable sitting. A vertical wall of size 39in X 19in is placed in a depth of 18in so that it can be reachable by an average adult sized person. BendDesk employs Frustrated Total Internal Reflection (FTIR) [38] to detect touches on the surface. For additional high precision input, an Anoto digital pen is used, which determines its position using a dot pattern that is printed on the diffusor and sends it out via Bluetooth. The BendDesk setup can be seen in figure 11.

**Curve :** The curve [99] is an improved shape for digital desks and takes into account ergonomic requirements described in the previous section and offers novel interaction possibilities. The Curve [99] consist of a horizontal and vertical interactive surface, seamlessly connected by a curved segment. The curve represents the state of the art design for a desk blending horizontal

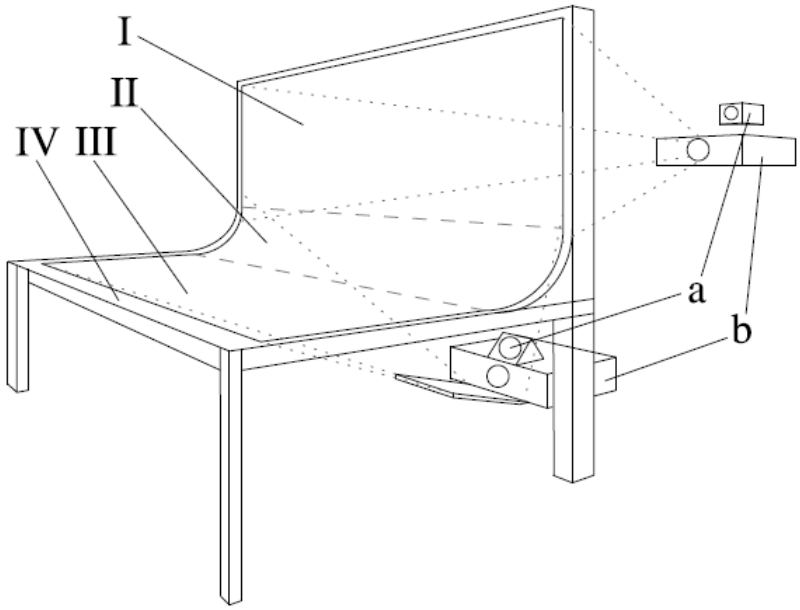


Figure 11: *BendDesk hardware setup with the following interactive areas: (1) Wall, (2) Curve, (3) Tabletop, (4) non-interactive strip. a) IR cameras. b) Projectors.*

and vertical surfaces. The vertical segment of The Curve panel was tilted backwards by 15 ° to reduce the strain on finger and hand as the finger can rest on the surface. The top edge of the vertical surface is 44cm above the horizontal surface and 5cm below the average user's eye level, allowing user to easily avert view from the screen. This allows the user to refocus at a distance object from time to time, reducing visual strain. The horizontal segment was designed to have a depth of 35cm, which is the maximum depth that still allows an average size user to comfortably reach the whole vertical segment. Also, the curved segment was designed to be 10cm, which offers a smooth transition between the vertical and horizontal segments (see figure 12 ).

Wimmer et al. [99] used a curved 12mm thick acrylic panel in the realization of their prototype, to get a seamless output and to preserve the possibility to use IR-based multi-touch input. Two projectors, each with a resolution of 1920 x 1080, were installed for back-projection on the screen. FTIR [38] was chosen for sensing multi-touch inputs and chains of SMD LEDs were assembled on the outer edge of the acrylic panel. Four Point

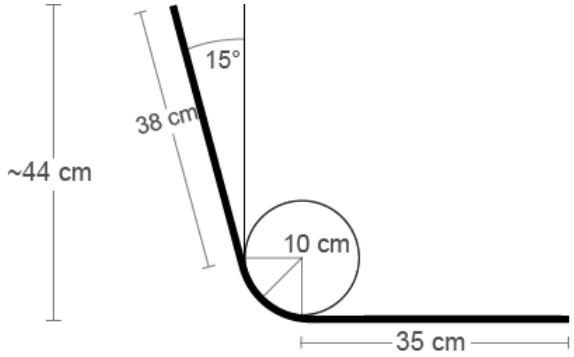


Figure 12: *Final Panel dimensions of The Curve.*

Grey Research FireFly MV cameras, each with a resolution of 640 x 480 px at 60Hz were used for tracking touch points (see figure 13).

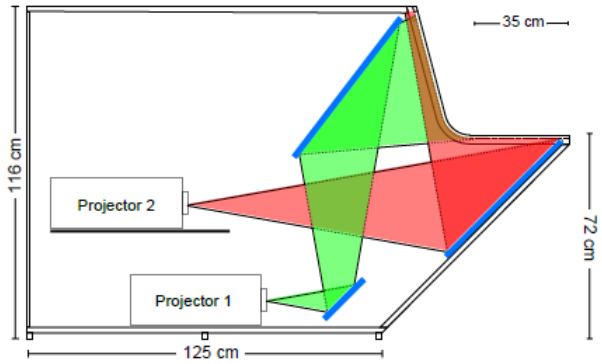


Figure 13: *Set up of The Curve*

The Curve [99] prototype design also had four major limitations.

1. **Screen Size and Resolution:** Although the current Curve prototype visual resolution of 1920 x 1730 px projected by two projectors onto a 90 x 80 cm area is sufficient for many office applications, it is insufficient for most reading tasks. Therefore a higher resolution would greatly improve the usability of the Curve.

2. **Leg Room :** The leg room is the amount of space between the floor and the horizontal surface of the desk where the user's legs rest. The

leg room of the Curve is limited in order to project on the horizontal surface at a perpendicular angle.

3. **Adjustability :** The nature of the setup of the Curve makes it rigid and impossible for users to adjust properties like inclination, height, or depth.

4. **Only Touch Sensing :** The FTIR used for tracking touches can not detect hovering gestures above the surface and makes it impossible to capture paper documents that are placed onto the surface.

### 2.6.2 Digital Desks with Interactive Surfaces

A digital desk can also be implemented using an interactive surface [26] [99] [98] [8]. A few of these digital desk implementations are briefly described below.

**DocuDesk :** DocuDesk [26] is an interactive desk that demonstrates interaction techniques for establishing many-to-many linkages between paper and digital documents. The DocuDesk augments a user's existing PC set up by including a Wacom Cintiq 21UX display, laying flat on top of the user's desk, bathed in IR light from above and filmed by an overhead video camera with IR filter. The Cintiq can be operated with a stylus and can run on a multi-monitor configuration with a user's existing PC setup. Paper placed above on top the DocuDesk is observed by the system's camera which uses standard vision algorithms to recognize several standard paper sizes. The system checks to see if a recognized paper has a 2D barcode and if so, tries to match it with its electronic counter part. If no bar code is found on the recognized paper, the system takes a snap shot of the paper and stores it in the system. An interactive desk enables a juxtaposition of physical paper and digital input and output. Paper placed on top of the surface can be augmented by displaying an interactive shadow menu next to the document to simplify common tasks such as annotation.(see Figure 14).

**Magic Desk** The Magic Desk [8] was designed to bridge multi-touch technology into desktop computing to give users additional input channels in their daily tasks. The design of the Magic Desk was a product of a series of studies done by Bi et al [8] which systematically evaluated the various potential regions with the traditional desktop configuration that could become multi-touch enabled. Guided by the study results, the Magic Desk implemented a set of interaction techniques integrating multi-touch with a mouse and keyboard to facilitate desktop work. The Magic Desk was implemented on a Microsoft Surface with a Dell multi-touch display. A keyboard and wireless mouse having tags on them were used so that their position and

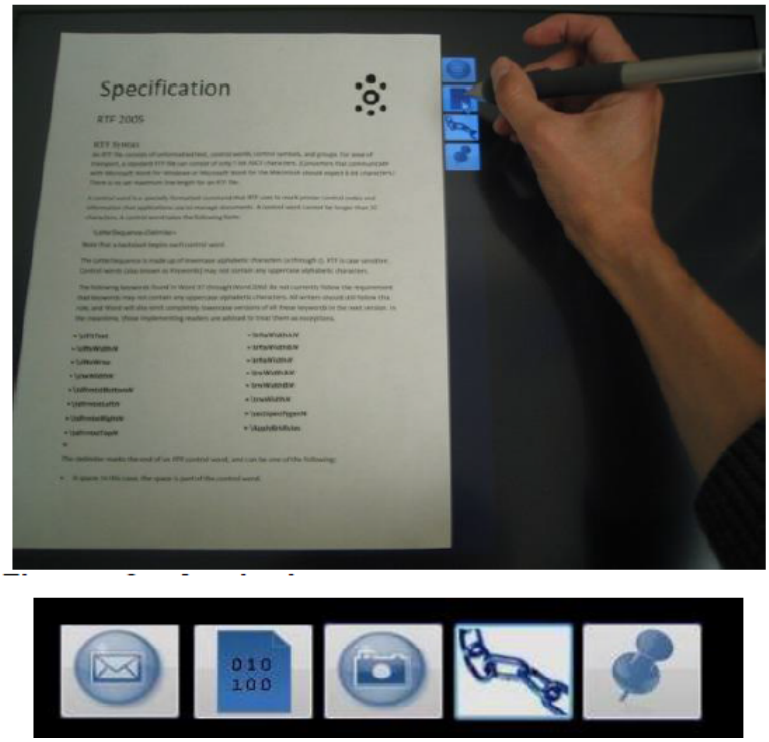


Figure 14: *DocuDesk: (top) desk surface, (bottom) interactive shadow menu*

orientation could be recognized by the surface. The Magic Desk was designed with three major components providing different functionality.

1. **Enhanced Task Bar:** This was designed to enhance flexibility and increase the input bandwidth of managing windows. The wider aspect ratio of the Enhanced Task Bar makes it possible for overlapping windows to be spread out more horizontally and thus are more accessible for manipulation. The Enhanced Task Bar enables the re-size, maximize/restore, minimize/restore operations using two handed interactions.

2. **Multi-Functional Touch Pad :** The multi-functional touch pad was designed to be positioned on the left hand side to enable two-handed interaction tasks such as controlling multiple degrees of freedom. The right hand interacts with the mouse while the left hand uses the touch pad. The touch pad serves a wide range of functionality including serving as a repository for storing commonly used UI items and also to adjust the control distance gain of the mouse.

3. **Digital Mouse Pad:** This was designed to augment mouse operations. Commands can be triggered by directly touching a button on the right-click mouse menu that is persistently visualized on digital mouse. bringing the common commands on the digital mouse pad allows users to quickly access them.

All the above mention digital desk implementations lack the ability to track paper documents in a pile. However, there has been some research [46] [74] [83] in the design of a digital desk system which tracks paper documents in a pile. In the next section, we will take a look at the need to research on paper documents in piles and review a few digital desk designs that attempt to address the tracking of paper documents in a pile.

### 2.6.3 Paper Documents in Pile

As stated in the previous section, people organize paper documents on their desks in a pile [58]. Many solutions have been proposed to bridge the gap between the paper and electronic documents, but they mainly focus on digitally incorporating paper annotations and lack the ability to track the document's physical location in a pile [46]. Over the past years, the use of different tracking and ID technologies such as barcodes, IR tags and RFID have become prevalent in the context of finding objects [94] [72]. These techniques could be applied to paper document tracking on the physical desktop but they require the use of physical tags and specialized readers for their implementation. Using video-based tracking techniques, the use of specialized tags could be avoided. There exists research on video-based tracking [60] [63], but they do not support tracking papers in stacks.

To keep track of documents in a pile via vision-based techniques, there has to be automatic analysis of the video frames of the desk to monitor whether a new document has been added or removed from the desk, and locating the pile which the document has been added to or removed from. Siio *et al* [83] tried to track documents in a cabinet drawer by using a camera to get a snap shot of the top document of the pile in the drawer, and using a laser to keep track of the height of the pile. This approach of using lasers to determine the height of the pile has the fundamental assumption that all the documents in the pile have a reasonable and equal height which can be measured with great precision. This cannot be implemented with paper documents because paper documents are relatively thin and changes in the pile height will not be significant enough to be detected by the laser.

The Self-Organizing Desk [74] also tried to track paper documents in a stack by constantly surveying the desk top, analyzing every video frame to detect changes on the desk surface and then keeping a history of the documents found on the desk. The change can be the addition of a paper document on the desk, the removal of a paper document, or a shift of an entire pile of paper documents. The Self-Organizing Desk [74] system tries to accommodate piles as follows. If the system notices a new change (for example adding a document to a pile), the approximate coordinates of the paper document are computed and the camera is automatically positioned so as to capture a picture of maximum details of the added document. This image is then filtered and the filtered data is indexed in a database. The database contains the layout of each document and a history of the orders in which documents arrive on the pile. An overview of the architecture of the Self-Organizing Desk is shown in figure 15.

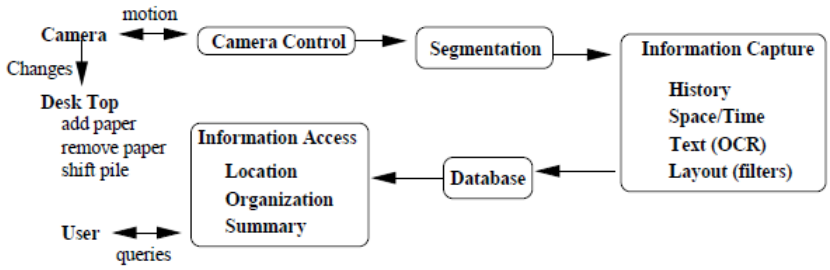


Figure 15: *Self-Organizing Desk architecture*

A change is detected by the the Self-Organizing Desk [74] by a segmentation module. The segmentation module consists of a four step algorithm. the first step compares basic images taken when the last event was detected against the current image, and the area change is extracted from the image. The second stage generates a new image of maximal details that contains the entire area identified. The third step identifies the enclosing border of the area of interest, by using object features and statistics on pixels. The final step parses the identified borders of interest to identify the pages. However, the Self-Organizing Desk [74] has the constraint that the paper documents must be of know size and are only allowed to translate.

Kim *et al* [46] also designed a system for tracking paper documents in a pile. The system captures the movement of document on the desk with an

overhead camera. The paper is then analyzed using SIFT [55] to link each paper document with its electronic copy on disk and track its physical location in the pile. Kim *et al* [46] detected a change in the scenery of the desk surface by calculating the difference between successive video frames and uses the SIFT algorithm to detect if a new paper document was added to a pile, removed from a pile or changed position from one pile to another. A scene graph was used to represent the occlusion order between paper documents and the scene graph is automatically updated if a paper document changes position. The system designed by Kim *et al* [46] had the limitation that only paper documents at the top of the stack can be moved. The system fails to keep track of the paper document if a document is arbitrarily removed from the middle of the pile.

The above mentioned digital desk solution track paper documents in a pile to some extent but also have some limitations. In some cases [46], for the system to work, only one document can move at a time and only the top most document can move. Other cases are not suited for paper documents because they use techniques such as laser technology to determine the height of the pile [83]. Also, in some other cases [45] the paper documents have to be of distinct appearances. The goal of this thesis is to design a digital desk system for tracking paper documents in piles using purely vision-based techniques that relax the assumptions and limitations of the current proposed digital desk systems for paper document tracking. Since this thesis uses purely vision-based techniques for tracking, the next chapter will review the current research trends and techniques for video-based document processing.

# 3 Video-Based Document Processing Methods

Processing a video frame for document recognition and identification entails processing the text in the document contained in the video. Unlike document capture using scanners where the images are of higher resolution and easily processable for text detection, automatic text document extraction and recognition in video frames is somewhat challenging because text present in video frames are of different sizes, orientation, noise, low resolution and contrast. Text in video frames can be classified into *"graphic text"*, which is text information which is artificially superimposed on the image (such as subtitles in news, sports scores etc.) and *"scene text"* which is text which naturally exists in the video frame (such as the text within a document from a video frame) [23]. The processing of scene text has some additional complexities such as being multi-orientated and multilingual [75]. A typical text frame processing system can be seen in Figure 16. Text frame selection determines whether a frame contains text information. text detection and localization finds and defines the actual location of the text present in the video frame. The text extraction stage simply extracts and binarize the localized text for the Object Character Recognition stage.

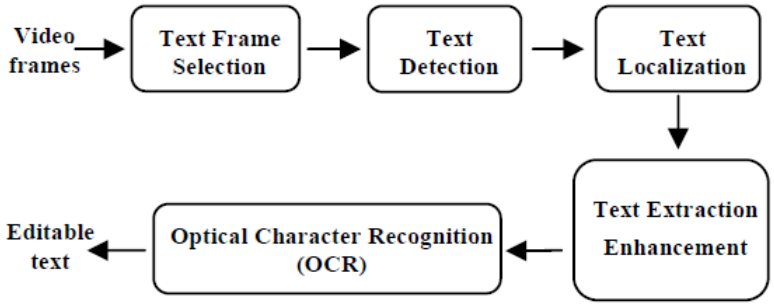


Figure 16: *A typical video processing system*

This section presents a review of various state-of-the-art techniques proposed by researchers towards different stages of text information processing in video frames and the next section will present a background study on Object Character Recognition (OCR).

## 3.1 Text Frame Selection and Classification

Text frame selection determines whether a frame contains text or not, before the process of text detection and recognition takes place. An efficient text frame selection method helps to avoid computationally expensive text detection methods on non-text frames present in the video as text detection algorithms can detect text incorrectly from non-text frames. In most of the text frame selection researches [62] [81] [77], the text frames are assumed to have information present in them.

The first method to be explored on text frame selection is the method proposed by Shivakumara *et al* [77]. This method works at the block level rather than at pixels level of a whole video frame, which ensures less time consumption in processing. The reason for choosing block level processing is that text in a video frame generally does not occupy the whole frame but rather occurs in small clusters of text. The method is based on wavelet and median moments in K-means clustering to classify probable text blocks among a set of 16 blocks of the frame. From the probable text blocks, the method uses the same wavelet and median moment's features with a MaxMin clustering method to choose probable dominant text pixel (PDP). Then four quadrants are formed for the selected PDP at the centroid of the pixels in the block. Next, for each quadrant, the pixels percentage is computed. Mutual nearest neighbor based symmetry (MNNS) [18] is used here to identify the presence of text using the percentage values of the four quadrants. If the method finds at least one true text block then the frame is considered as a text frame. Otherwise it is considered as a non-text frame. Haar wavelet [16] is used for decomposition followed by median moments to detect probable text blocks and probable text pixels. Wavelet decomposition, good in enhancing edge pixels by suppressing low contrast pixels in the background and the median moments are good in extracting variations in the intensity value and spatial distribution of pixels [79]. The combination of these two features helps in classifying text and non-text pixels accurately as text and non-text pixels differ in intensity values and their spatial distribution.

The second method to be explored was proposed by Na and Wen [62], which is a multilingual video text tracking algorithm based on the extraction of Scale Invariant Feature Transform (SIFT) [55] descriptors through video frames. The proposed method also consists of a global matching technique using geometric constraints to decrease false matches of the SIFT descriptors which effectively improves the accuracy and stability of text tracking results. Based on the correct matches, the motion of text is estimated in adjacent

frames and a match score of text is calculated to determine the Text Change Boundary (TCB). A flowchart demonstrating the method can be seen in Figure 17. The method consists of five steps: SIFT feature extraction, SIFT feature matching, global matching based using geometric constraints, motion estimation and TCB determination.

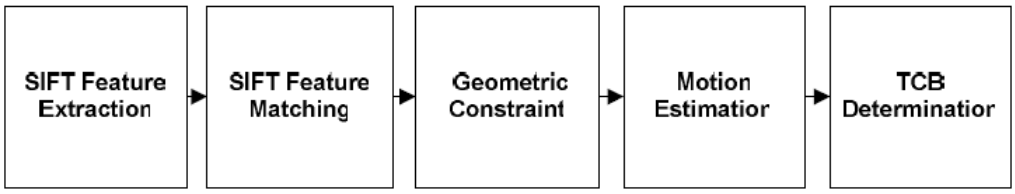


Figure 17: *Flowchart of video text tracking method by Na and Wen [62]*

*SIFT Feature Extraction and Matching :* The feature extraction and matching step is done in three phases.

1. Extract SIFT feature points in reference text box.
2. Extract SIFT feature points in the candidate region which is larger than the reference text box by adding 80 pixel to the length and 40 pixel to the height of the reference text box respectively in order to estimate text motion.
3. After SIFT feature extraction, keypoint matching between two sets of points is implemented using the nearest neighbour algorithm [55].

Unlike the original implementation of the SIFT algorithm [55] where the threshold of the distance ratio is 0.8, Na and Wen [62] investigated the relation between the threshold of distance ratio and ratio of correct matches and found that a threshold of 0.65 works better in eliminating false matches.

*Global Matching Geometric Constraint:* This step deals with the further discarding of false matches in the video frames by using Geometric Constrains (GC) on candidate matches found by the nearest neighbour algorithm in the previous step to filter out correct matches. Geometric constraint can be described by the Euclidean distance among keypoints. If a keypoint has a different relative location to other keypoints in the current frame from that of its correspondence in the reference frame, this keypoint is probably falsely

matched and is therefore discarded. By setting the error threshold of relative position, false matches are eliminated and correct matches filtered.

*Motion Estimation Based on Global Matching:* In this step, four different parameters namely two shifts, one rotation angle and a zoom factor are used to describe inter-frame motion. The feature points in the reference frame $(x_p, y_p)$ are associated with the feature points in the current frame $(x_c, y_c)$ with the transformation in the following equation.

$$\begin{bmatrix} x_c \\ y_c \end{bmatrix} = \begin{bmatrix} \lambda\cos\theta & -\lambda\sin\theta \\ \lambda\sin\theta & \lambda\cos\theta \end{bmatrix} \begin{bmatrix} x_p \\ y_p \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

Where $\theta$ denotes the rotation angle, and $\lambda$ the zoom parameter, $t_x$ and $t_y$ respectively X-axis and Y-axis shifts. Because the wrong matches have been already discarded by geometric constraint, a linear Least Squares Method [12] was applied on the set of corresponding points that were obtained from matching process to achieve better accuracy.

*Detection of Text Change Boundary:* Text Change Boundary (TCB) refers to the boundary frame of variation of text content. When TCB appears, the video frame text usually changes completely, which means there are no correct text matches. The detection of such boundaries to grasp the entire text event in a video frame is very important. In these cases, the geometric constraint can effectively discard these false matches and detect the text change by detecting less correct matches than threshold. However, in some cases, the texts in new frames may not be totally different from the TCB. For example, the same characters or words appear in new texts in consecutive frames. The keypoints in these reappeared characters will be mistaken for correct matches by geometric constraint even in different semantic texts. So, using geometric constraint alone cannot faithfully determine TCB. To solve this issue, Na and Wen [62] divided the reference text box and candidate text box equally into $m$ subregions. $m$ is calculated according to the ratio of its length and height. In each subregion, the gradient orientation bins as the feature describing the region. To describe the similarity of subregion $i$ between the reference text box and candidate text box, match value is defined as:

$$H_i(R,C) = \sum_{j=1}^{8} min(R_j, C_j) / \sum_{j=1}^{8} R_j$$

where $H_i(R,C)$ is the match value, and R and C are reference subregion and candidate subregion histograms respectively, each containing eight orientation bins. The match value is computed for every model histogram and the value is closer to unity if the model image is more similar. It is obvious that a match value of unity is obtained for an image compared with itself. With the match value of each subregions, a match score (MC) which describes the global similarity between the reference text box and candidate text box is computed:

$$MC = \sum_{i=1}^{m} H(R,C)/m$$

where MC is the match score of the two text boxes and $m$ is the number of subregions. The tracking process continues if and only if MC is larger than a certain threshold. If not the candidate frame will be determined as a TCB.

## 3.2 Text Detection and Localization

Text detection and localization can be divided into two categories, namely, Region-based and Texture-based techniques.

1. **Region-based techniques:** Region-Based techniques work in a bottom-up fashion, by dividing the frame into small regions to form bounding boxes for the text.

   The first region-based method to be explored was proposed by Dinh et al [22], which is an effective technique for text detection in video based on the similarity in stroke width of text (which is defined as the distance between two edges of a stroke). From the observation that text regions can be characterized by a dominant fixed stroke width, edge detection with local adaptive thresholds is first devised to keep text while reducing background-regions. Second, morphological dilation operator with adaptive structuring element size determined by stroke width value is exploited to roughly localize text regions. Finally, to reduce false alarm

and refine text location, a new multi-frame refinement method is applied.
Another region-based technique proposed by Jung et al [44], was a stroke filter based method. Based on a stroke filter response and text polarity, local region growing was used to segment the text. An Optical Character Recognition (OCR) feedback score was used to improve text segmentation accurately. The method can broken down into four steps. An overview of the method can be seen in figure 18.

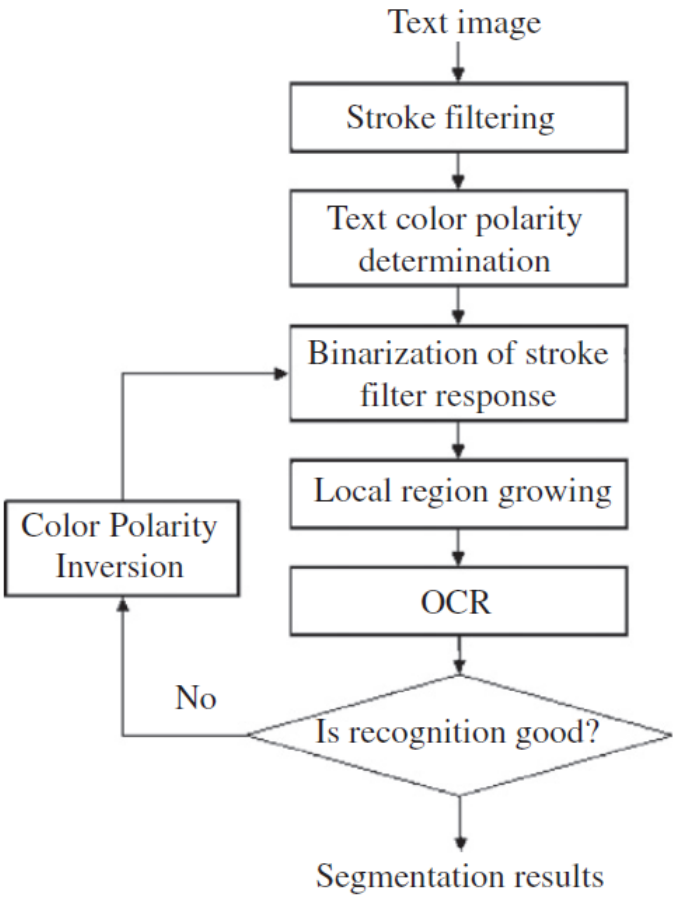


Figure 18: *Overview of the stroke filter based technique for text segmentation*

(a) *Stroke Filtering:* This is the first step of the method and it is responsible for filtering strokes in the video frames. A stroke is defined as a straight line or arc used as a segment of text and

the texts in videos comprise several strokes [44]. A stroke filter is designed based on the definition that a video frame segment is defined as a text if and only if several stroke-like structures exist in it. In order to design the stroke filter, a local image region is first defined to be a stroke like structure if: it is different from its lateral regions, intensities of its lateral regions are similar, and it is nearly homogeneous with respect to its intensities. The next procedure of this step is to obtain the bright and dark stroke filter responses ($R^B$ and $R^D$ respectively) of all the pixels $(x, y)$ within a text region of a video frame. This was done as shown:

$$R^B_{\alpha,d}(x,y) = \frac{\mu_1 - \mu_2 + \mu_1 - \mu_3 - |\mu_2 - \mu_3|}{\sigma}$$
$$R^D_{\alpha,d}(x,y) = \frac{\mu_2 - \mu_1 + \mu_3 - \mu_1 - |\mu_2 - \mu_3|}{\sigma}$$

Where $\mu_i$ denotes the estimated mean of the intensities in the region $i$. $d$ is the width of a rectangular region in the video frame containing text. $\sigma$ denotes the standard deviation of intensities in a region and it is a measure of the extends to which the intensities of the region are spread out. After obtaining the filter responses, the stroke features $(R_BO_B, S_B, R_D, O_D, S_D)$ of any pixel $(x, y)$ is extracted using the following expressions.

$$R^B(x,y) = \max_{(\alpha,d)} R^B_{\alpha,d}(x,y),$$

$$O^B(x,y) = arg \max_{(\alpha)} R^B_{\alpha,d}(x,y),$$

$$S^B(x,y) = arg \max_{(d)} R^B_{\alpha,d}(x,y),$$

Where R, O, S respectively denote the response, orientation and scale of the stroke filter whereas B and D denote the bright and dark stroke filters, respectively.

(b) *Text Colour Polarity Determination:* In order to determine the text colour polarity automatically, a bright and dark stroke filtering is first performed to obtain $R_B$ and $R_D$. Using $R_B$ and $R_D$, two features, $F_R$ and $F_E$ for the determination of text colour polarity can be obtained as follows:

$$\frac{\Sigma_{(x,y)} R^(B)(x,y)}{\Sigma_{(x,y)} R^(D)(x,y)}$$

$$F_E = \frac{N^(B)}{N^(D)}$$

Where $F_R$ is the ratio of the sums of the magnitude of the bright and dark stroke, and $F_E$ is the ratio of the sums of the number of edge points in the binarized response maps of the bright and dark stroke filters. $N^(B)$ and $N^(D)$ denote the number of edge points in the binarized map of the bright and dark stroke filter responses, respectively. Using a support vector machine (SVM) classifier with a radial basis function kernel, the text colour polarity is accurately determined.

(c) *Local Region Growing:* The purpose of this step is to recall the pixels missed in the binarized filter response map, for accurate and robust text segmentation. The procedure is based on the stroke filter response and combines a global Probability Density Function (PDF) and local similarities to achieve a reliable performance. The local region growing algorithm used is described below:

*Input:* I - initial segmentation result (binarized stroke filter response map obtained in step (a); S - source text image
*Step 1:* For I and S, estimate PDF of text colour
*Step 2:* For each white pixel in I, if the number of white pixels in its 3X3 neighbours is written [53], then go to step 3, else step 2
*Step 3:* for each pixel in the 3 X 3 regions, if it is: (1) similar to its text neighbours and (2) of high probability according to the PDF, then it is marked as text. Repeat steps 3 and 2, until no pixel is changed.
*Output :* Refined segmentation result.

(d) *Feedback from the OCR module:* The purpose of this stage is to improve the accuracy of the determination of text colour polarity and the performance of text segmentation, by applying an additional verification step for the segmentation result of the local region growing procedure by an OCR module. The feedback of an average recognition score of characters of the OCR module is used to do the additional verification. The average recognition score, $S_A$, can be calculated as follows:

$$S_A = \frac{1}{N} \sum_{i=1}^{N} S_i$$

$$S_i = K_G x S_G + K_G x S_G,$$

where $i$ and $N$ denote the index and the number of characters, $K_G$ and $K_R$ are constants and $S_i, S_G$, and $S_R$ denote the total, geometric, and recognition scores of each character respectively.

Shivakumara et al [78] proposed a novel technique for detecting both graphic text and scene text in video images by finding segments containing text in an input image and then using statistical features such as vertical and horizontal bars for edges in the segments, to detect true blocks efficiently. In their research [78], a frame was segmented into 16 non-overlapping blocks. Mean and median filter, and edge analysis were used to identify the candidates text blocks, and the complete text block was obtained using block growing method.

Anthimopoulos et al. [1] proposed a hybrid system for text detection in video based on the edges, local binary pattern operator and SVM. The system consists of two main stages. The first stage detects text regions based on the edge map of an image leading in a high recall rate with minimum computation requirements. The second stage is a refinement stage which uses an SVM classifier trained on features obtained by a new Local Binary based operator which results in diminishing false alarms.

A method which uses temporal information for moving text detection was proposed by Huang et al. [41]. The temporal information is obtained by dividing a video frame into sub-blocks and calculating an inter-frame motion vector for each sub-block. Text blocks are extracted from the sub-blocks through inter-frame spatial relationship checking and inter-frame classification. Their method [41] is robust towards low resolution and complex backgrounds, and it works well on on detecting scrolling text in news clips and movies. An overview of their method can be seen in Figure 19.
Another method proposed by Zhang and Sun [106] uses a Pulse Coupled Neural Network (PCNN) edge based method for locating text. The PCNN is used to generate a firing map and to segment an image into different planes and detected edges using the already generated firing map and a phase congruency detector. Zhang and Sun [106] made tests on a large dataset to evaluate their method which efficiently detected text with various colors, font sizes, positions, and uneven illumination. A summary of their method is shown in Figure 20.

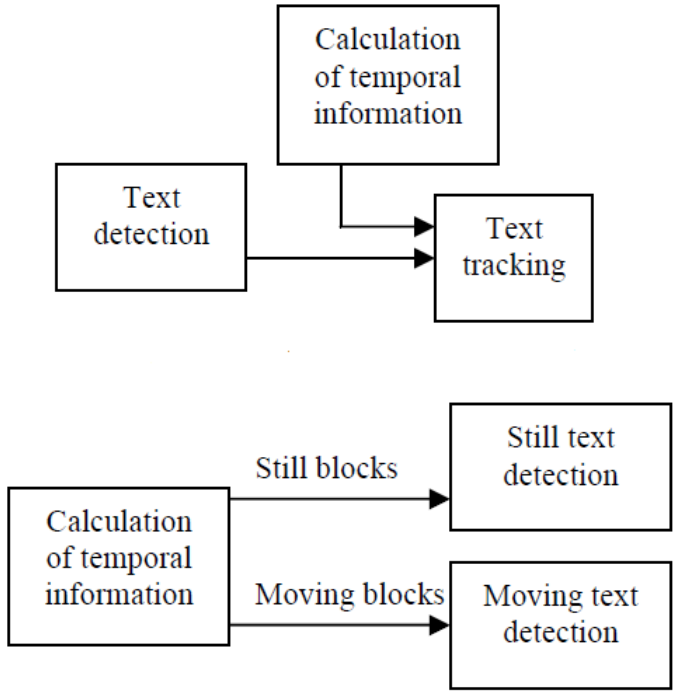


Figure 19: *temporal information for tracking(above), temporal information for detection(below)*

Recently, Uchida st al. [91] established the Speed Up Robust Features(SURF) method which can be used to detect character regions and to distinguish text and non-text regions with good accuracy. A more detailed description of the SURF algorithm will be done later in the chapter on implementation.

2. **Texture Based Methods:** Most of the recent works on texture based detection and localization are based on wavelet transforms and its variations [103] [79] [43] [107].
   Ji et al. [43] proposed a method which entails the use of two texture features namely wavelet coefficients and Gray-level co-occurence matrix from text detection along with SVM. First, a small overlapped sliding window is scanned over a video frame from which hybrid features are extracted. This is followed by employing an SVM classifier to distinguish the text from background. Lastly, a vote mechanism and morphological filter are performed to precisely locate the text region. Four different kinds of video were evaluated with this method [43] and

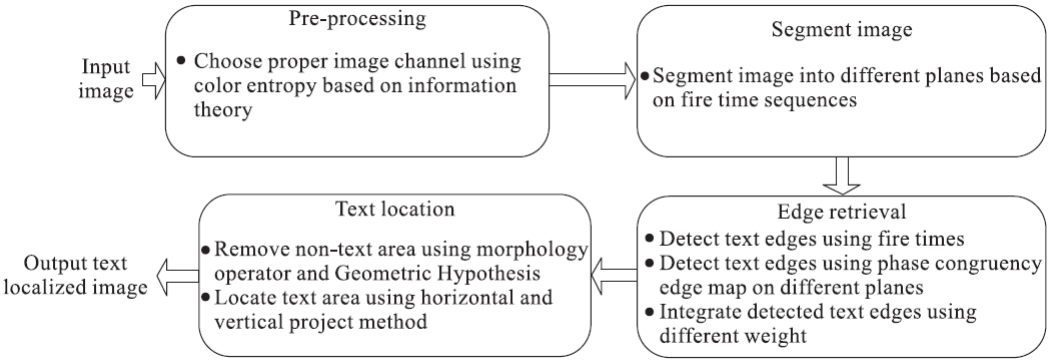


Figure 20: *Flowchart of the PCNN algorithm*

the experiments showed its high performance.

Also, Zhao et al. [107] used a wavelet transform and space representation with discriminative dictionaries for text detection. In their proposed method, the edges of a video frame are detected by the wavelet transform and scanned into patches by a sliding window. By applying a simple classification procedure using two learned discriminative dictionaries, the candidate text areas are then obtained. Finally an adaptive run-length smoothing algorithm and projection profile analysis are used to further refine the candidate text areas. This method proposed by Zhao et al. [107] was evaluated on the Microsoft common set, the ICDAR 2003 text locating set, and an image set collected from the web. The tests produced an effective detection of text of various sizes, fonts and colours. The flowchart of the proposed method can be seen in Figure 21.

In 2011, Shivakumara et al. [80] proposed a Laplacian approach for muti-oriented text detection in videos. An input video frame is filtered with Fourier-Laplacian followed by a K-means clustering to identify candidate text regions based on the maximum difference. The skeleton of each connected component helps to separate the different text strings from each other. Lastly, text straightness and edge density are used for false positive elimination. This proposed method [80] is able to handle graphics text and scene of both horizontal and non-horizontal orientation. The flow chart for the method can be seen in Figure 22.

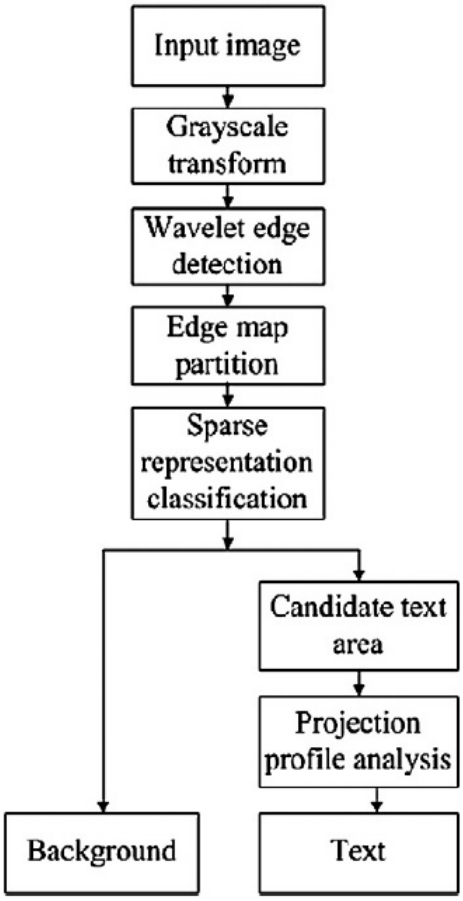


Figure 21: *Flowchart of text detection method*

Peng et al. [69] proposed a Conditional Random Field (CRF) based approach to detect the text lines from video frames. The proposed system contained three basic stages as shown in Figure 23: text block extraction based on edges, support vector machine (SVM) prediction and CRF labeling for text regions, and the text line aggregation.

## 3.3 Extraction, Binarization, and Enhancement

Extraction and binarization, often used synonymously, aim towards the extraction of individual characters from detected and localized text blocks for Optical Character recognition (OCR). a wide range of binarization techniques have been used by researchers in the last few

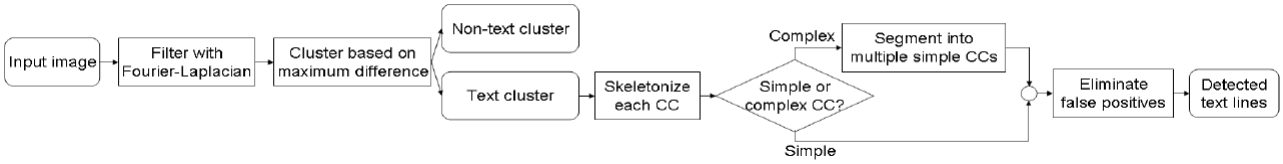


Figure 22: *A laplacian approach to multi-oriented text detection in video (CC = Connected Components)*


Figure 23: *Conditional random field approach for text detection*

decades [54] [54] [76] [70] [65] in order to get an enhanced image. Some of these techniques are briefly described in this subsection.

### 3.3.1 Character Extraction and Segmentation Techniques

A tensor voting based text segmentation technique was proposed by Lim et al [54] in which a video frame (which can be grayscale, pure colour, or a mix of grayscale and colour) is first of all decomposed into chromatic and achromatic regions. Using tensor voting and adaptive median filter, text layers were identified and noise removed from the video frame. Finally, density estimation for center modes detection and K-means clustering algorithm was performed for the segmentation of values according to hue or intensity component in the improved image. The overall framework of the method proposed by Lim et al [54] is shown in Figure 24.

Shivakumurara et al. [76] proposed a gradient based character segmentation scheme in which the Bresenham's line drawing algorithm [85] was used to handle multi-oriented text for the extraction of gradient features. Min-Max clustering was used to separate text and non-text cluster and segmentation was achieved based on the height difference, top distance and bottom distance with a vector union operation.
Furthermore, a Gradient Vector Flow (GVF) [102] based method was proposed by Plan et al. [70] in 2011. the proposed technique [70] al-

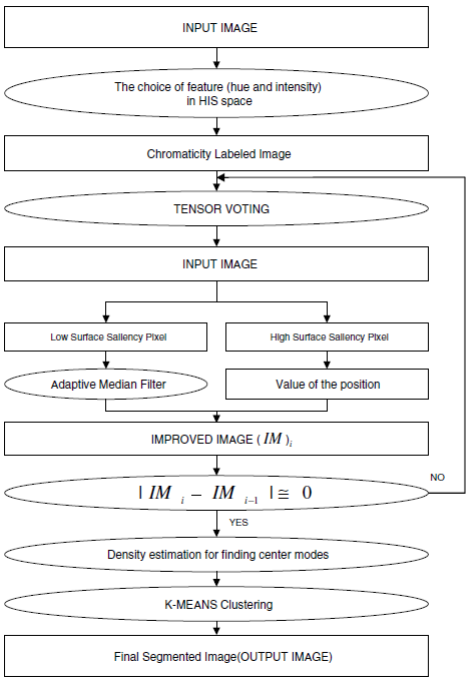


Figure 24: *overall framework of a tensor voting based text segmentation method*

lows curved segmentation paths and thus is able to segment overlapping characters. GVF was used here to identify candidate cut pixels. A two-pass path finding algorithm was then applied where the forward direction helps to locate potential cuts and the backwards direction serves to remove the false cuts.

### 3.3.2 Binarization Techniques

Ntrirogiannis et al. [65] proposed a binarization method based on the detection of the text baselines in order to define the main body of the text. The stroke width of the characters detected from the main body of the text was used to address a two step binarization process. At the first binarization step, different valuation parameters are used for the inside and outside area of the main body of the text. A convex hull analysis [4] is performed to include thin or broken binarized parts that may exist outside the main text body. At the second step, binarization

is performed with different valuation in parameters for the inside and outside area of the entire text body. Figure 25 shows the flow chart for the proposed binarization technique [65]

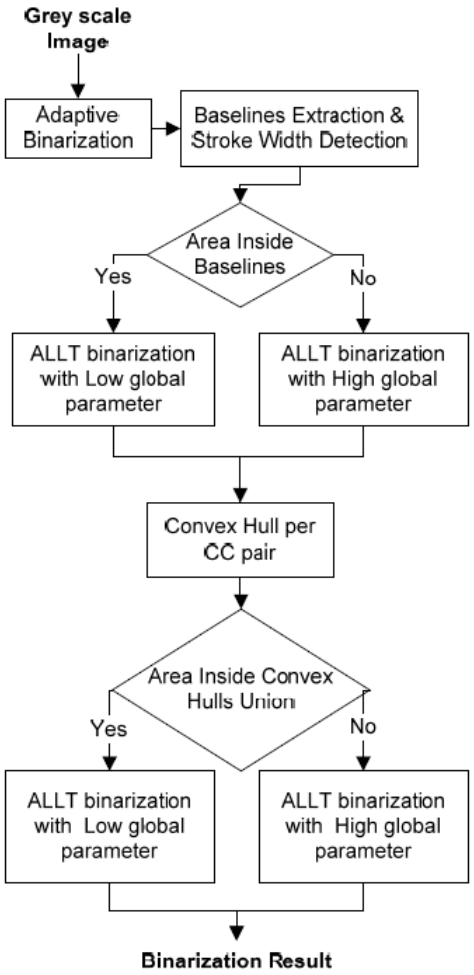


Figure 25: *Technique for text binarization in video frame*

## 3.4 Optical Character Recognition

The main purpose of Optical Character Recognition (OCR) is to deal with the recognition of optically processed characters [24]. The process of OCR can be performed both off-line and on-line. The offline case is the recognition of characters after the writing or printing has been completed while the on-line case is the automatic recognition of characters as they are drawn.

A generic OCR engine consists of several components which can generally be grouped into 5 major components [24] namely: Optical Scanning, Location Segmentation, Pre-Processing, Feature Extraction and Recognition Post processing (see figure 26 for a flow diagram).

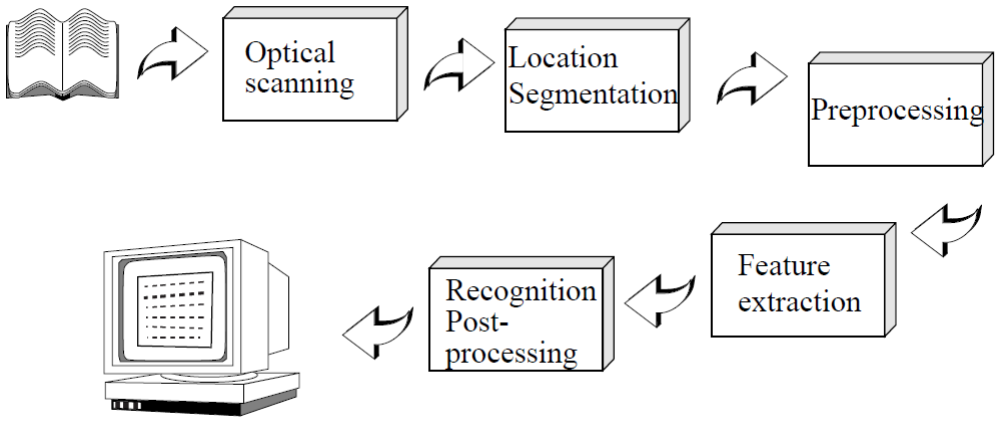


Figure 26: *Components of a typical OCR system*

1. *Optical Scanning:* This component deals with capturing the digital image of the original document and converting the light intensity of the image into a bi-level image of black and white. This process is know as thresholding, which is performed to save memory space and computational effort. The result of the final recognition depends highly on the quality of the bi-level image. Typically, as in the case of most OCR systems [13] [5] [84] a fixed threshold is used where the gray-levels below this threshold are assigned to be black and levels above are assigned to be white. In cases where the documents have a large range of contrast, a more sophisticated method for thresholding is employed [31] which is able to vary the threshold over the document, adapting to the local properties such as contrast and brightness.

2. *Location and Segmentation:* The segmentation process determines the constituents of an image. Segmentation is the step in which observed patterns in an image are segregated into units of patterns that seem to form characters [28]. In text, segmentation is the isolation of characters or words which is done by isolating each connected character component. In the 1970's, with the first-generation OCRs, segmentation was mostly done with the use of a flying spot scanner which used a cathode

ray tube (CRT) to focus light on a document which was gathered and sensed by photomultipliers. Only the image data for a single character area were entered into the recognition processor at one time because of the limited memory capacity in those days. The CRT method was later on replaced by the use of a laser image scanner consisting of a polygonal mirror, which scanned a paper document vertically over the height of one character. Horizontal scanning was also done by another mirror that changed angle once per character line. Using the laser scanner, OCR could only obtain pixel data from a single line of characters. Around 1977, a solid state image sensor, the Charged Couple Device (CCD) [42] replaced the laser scanning method. With the CCD, the vertical scan was done by the CCD itself, and the horizontal scan by mechanical movements of the CCD line sensors. In the early 1980's, with the second-generation of OCR's, with the availability of memory, full pages were scanned and stored in memory by a more sophisticated CCD line sensor with 4096 bits. The CCD and laser scanning methods for segmentation were pixel-oriented approaches which suffers from issues like distinguishing noise from text, the extraction of touching and fragmented characters, mistaking text for graphics or geometry and mistaking graphics or geometry for text. However, these issues could be resolved by using an advanced character segmentation method like the pattern-oriented segmentation method [28].

3. *Preprocessing:* Preprocessing is mostly responsible for noise reduction and normalization of the resulting image from the optical scanning and segmentation phase to obtain characters of uniform sizes. Noise reduction involves a combination of several techniques including smoothing, filtering, thinning, dehooking and stroke correction (for hand written characters). Smoothing usually averages a point with its neighbours. Some smoothing techniques [2] [32] average a point with just the previous points, allowing the computation to proceed as each point is received. Filtering eliminates duplicate data points and reduces the number of points in the images. Some filtering techniques [2] [32] [9] force a minimum distance between consecutive points, which produces equally spaced points. Furthermore, hooks that usually occur at the beginning and the end of a character stroke can be eliminated by dehooking algorithms [56] as well as stroke connection algorithms [15].

   Normalization methods aim to remove the variations of the writing and obtain standardized data. There are several basic methods for normalization [33] serving different purposes, some of which include

the following:

(a) *Skew Normalization and Baseline Extraction:* This deals with characters that are slightly tilted or curved within an image. Some methods of baseline extraction include using a form of nearest neighbour clustering [39], cross correlation methods between lines [17], and using Hough Transform [105].

(b) *Slant Normalization:* Slant normalization is used to normalize characters of different slant angle between longest stroke in a word and the vertical direction. There exist a number of slant normalization methods. Some of them extract vertical line elements from the contour of the character by tracing chain code components using a pair of one dimensional filter [57]. Other methods use an approach in which projection profiles are computed for a number of angles away from the vertical directions, where the angle corresponding to the projection with the greatest positive derivative is used to detect the least amount of overlap between vertical stroke, hence slant angle [34].

(c) *Size Normalization:* Size normalization is used to adjust the character size to a certain standard. Both horizontal [3] and vertical size normalization methods [48] can be employed by an OCR system.

4. *Feature Extraction:* Feature extraction involves the extraction of certain features that characterize a character while ignoring the unimportant attributes. An evaluation of some feature extraction techniques done by Eikvil [24] is shown in the figure 27. The criteria used in the evaluation is as follows:

**Robustness.**

1) *Noise:* Sensitivity to disconnect line segments, bumps, gaps etc.
2) *Distortions:* Sensitivity to local variations like improper protrusions, rounded corners, dilations and shrinkage.
3) *Style variation:* Sensitivity to variation in style like the use of different shapes to represent the same character or the use of serifs, slants etc.
4) *translation:* Sensitivity to movement of the whole character or its components.
5) *Rotation :* Sensitivity to change in orientation of the characters.

**Practical Use.**
1) *speed of implementation.*
2) *Complexity of implementation.*
2) *Independence:* The need of supplementary techniques.

| Feature extraction technique | Robustness 1 2 3 4 5 | Practical use 1 2 3 |
|---|---|---|
| Template matching | ● ● ○ ○ ○ | ○ ● ○ |
| Transformations | ○ ● ● ● ● | ○ ○ ● |
| Distribution of points: Zoning | ○ ● ○ ○ ● | ● ● ○ |
| Moments | ● ● ○ ● ● | ○ ● ○ |
| n-tuple | ● ○ ● ○ ● | ● ● ● |
| Characteristic loci | ○ ● ● ● ● | ● ● ○ |
| Crossings | ○ ● ● ● ● | ● ● ○ |
| Structural features | ○ ● ● ● ● | ● ○ ● |

● High or easy ● Medium ○ Low or difficult

Figure 27: *Evaluation of feature extraction techniques*

5. *Postprocessing:* As an image passes from one OCR stage to another, important information may be removed, since the context information is not available at the earlier stages (Optical scanning, segmentation, preprocessing and feature extraction). The lack of context information during the segmentation stage may cause even more severe errors since meaningless segmentation boundaries will be obtained. The purpose of the postprocessing stage is to bring context to the processed image and possibly, the incorporation of context and shape information to the earlier stages of OCR via a feedback mechanism. The simplest way to incorporate the context information is the utilization of a dictionary [11] for the purpose of spell checking. String matching algorithms can be used to rank the lexicon words using a distance metric that represents various edition cost [51].

# 4 Design and Implementation

So far we have looked at a few digital desk related works, and some vision-based techniques to process paper documents. In this section, the tools and algorithms used to accomplish the goal of recognizing, localizing, identifying and tracking paper documents on a physical will be explained.

## 4.1 Tools and Frameworks Used

### 4.1.1 OCR Tools

The first approach that was considered for document recognition, identification and tracking was to use a state-of-the-art OCR engine to recognize and identify the paper documents. This lead to the testing of a few OCR engines including Tesseract OCR engine [84], Asprise OCR engine [1], Microsoft Office OneNote [2], SimpleOCR [3] and ABBY [4]. These OCR engines were tested on paper document images. Several experiments were carried out using the different OCR tools mentioned, and with each experiment the paper document orientation, and resolution of the paper document images were varied. The Goal of the experiments was to verify the following: if an OCR engine could be used to recognize the presence of paper documents in an image frame and an if OCR engine could be used to recognize the characters within the paper document. Figure shows the sample images that were used.

The first set of experiments were aimed to test the accuracy of the OCR engines to recognize the texts in an image frame containing a single paper document. The images were taken with an HD (1080p) Microsoft LifeCam Studio digital camera mounted 60cm above a desk top surface. The document orientation, the light intensity into the camera and image resolution were varied for each tool to determine if changes in these factors affects the performance of the OCR engine. The images of 40 different paper documents were used in the experiments. The result obtained from the experiments were analyzed and the performance of the tools was evaluated based on the precision (the fraction of retrieved characters that are relevant) and Recall(the fraction of relevant characters that are retrieved). A summary of the experiments conducted can be seen in table 28. The second set of experiments that was conducted was verity if multiple paper documents can be recognized

[1] http://asprise.com
[2] http://office.microsoft.com/en-us/onenote/
[3] http://www.simpleocr.com
[4] http://www.abbyy.com

using the OCR engines.

| | Microsoft Office OneNote | Simple OCR | ABBYY | Asprise | Tesseract |
|---|---|---|---|---|---|
| **Oriented image in high light intensity (large font size)** | Poor results Precision =10/20 Recall =10/30 | Very bad results Precision = 1/40 Recall = 1/30 | Good results Precision = 25/29 Recall =25/30 | Average results Precision=15/35 Recall=15/30 | Good results Precision =27/30 Recall = 26/29 |
| **Oriented paper in poor light intensity (large font size)** | Very poor results Precision=1/26 Recall = 1/30 | Very Poor results Precision =0 Recall =0 | Poor result Precision =9/35 Recall =9/30 | Average results Precision=15/35 Recall=15/30 | Average results Precision =20/39 Recall =11/31 |
| **Upright image in high light intensity (large font size}** | Same results as oriented case | Same results as oriented case | Same results as oriented case | Same result as oriented case | Same results as oriented case |
| **Upright image with poor light intensity (large font size)** | Same results as oriented case | Same results as oriented case | Same result as oriented case | Same result as oriented case | Same results as oriented case |
| **Upright image average light intensity (regular font size)** | Average results Precision=35/37 Recall=35/80 | Poor results Precision=20/90 Recall=35/80 | Average results Precision=40/82 Recall =40/80 | Average result Precision =39/80 Recall= 39/80 | Good results Precision =85/90 Recall =80/90 |

Figure 28: *Experiment Results with OCR tools*

### Problems Encountered With OCR Engines

The purpose of experimenting with the OCR engine was to determine if OCR was the best computer vision solution for paper document recognition and identification. From the experiments conducted, it was realized that OCR tools are designed to recognize characters for an image with high resolution (scanned image). However, the project aims to recognize paper documents on a desk using an overhead camera which does not produce images of same resolution as scanned images. An example of the images produced by using an overhead camera and those from a scanned image of a paper document can be seen in Figure 29.

With images like in Figure 29, the OCR tools recognized very little characters and no characters in some cases from the paper documents in the images. However, the document orientation had little effect on the character recognition by the OCR engines.

After careful review, it was realized that depending only on OCR for recognition and identification was somewhat trivial because there are many factors like document's physical and logical structure [10], and pixel intensity variation that could be used for the recognition and identification of paper


Figure 29: *Difference in the image resolution of scanned images and images from overhead camera. Left: Scanned image, Right: Image from overhead camera*

documents. After careful research and consideration, the C++ implementation of the OpenCV (Open Source Computer Vision) library was used to accomplish the project.

### 4.1.2 The OpenCV Library

OpenCV is an open source library for developing computer vision applications. The OpenCV library contains over 500 optimized algorithms [52] for image and video analysis and is used world wide for different applications including interactive art, stitching maps on the web and advanced robotics. OpenCV was originally developed at Intel by a team lead by Gary Bradski as an initiative to advance research in vision and promote the developement of rich, vision-based CPU-intensive applications. The fact that the OpenCV library is an open source library makes it one of the main reasons why it was chosen for the implementation of this project. OpenCV is also one of the most stable and cutting edge implementation library for computer vision algorithms.

The OpenCV library is divided into several modules (see figure 30) with each module consisting of optimized state of the art computer vision algorithms for specific classes of computer vision technique. These modules include:

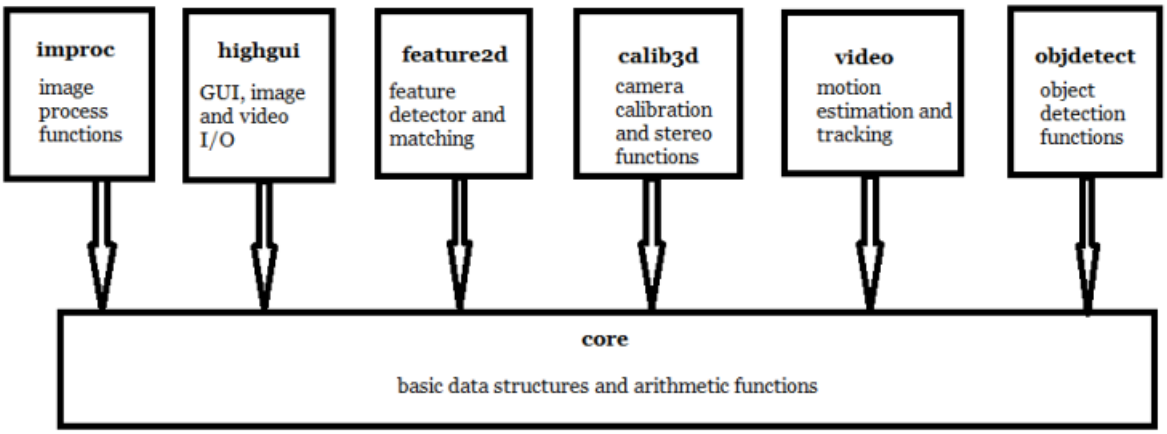


Figure 30: *The basic structure of OpenCV*

1. The *opencv core* module that contains the core functionalities of the library and the basic data structures and arithmetic functions.

2. The *opencv improc* module that contains the main image processing functions.

3. The *opencv highgui* module that contains the image and video reading and writing functions, along with other user interface functions.

4. The *opencv features2d* module that contains the feature point detectors and descriptors and the feature point matching framework.

5. The *opencv calib3d* module that contains the camera calibration, two-view geometry estimation, and stereo functions.

6. The *opencv video* module that contains the motion estimation, feature tracking, and foreground extraction functions and classes.

7. The *opencv objdetect* module containing the object detection functions such as the face and people detectors.

The C++ implementation of OpenCV 2.4.6 library and Microsoft Visual C++ 2010 were used to realize the project.

## 4.2 Project Setup

The setup for the project is simply a camera (an HD Microsoft LifeCam Studio - 1080p) mounted 60$cm$ above a desk surface so as to get the entire view of the working space of the desk surface. The camera is connected to a computer for image processing. The project setup can be seen in figure 31

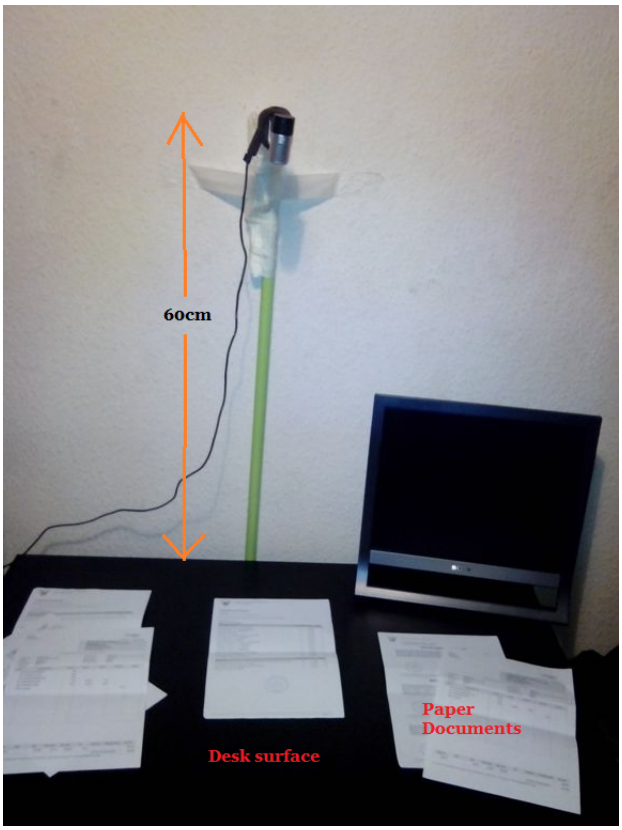


Figure 31: *Project Setup*

To accomplish the goal of the project, a top-down approach was used in the implementation. The video frames from the camera were processed in various stages to first recognize the paper documents, localize the recognized paper document, identify them and match each recognized paper document with a group of digital documents stored in memory.

### 4.2.1 Assumptions

A few assumptions were made for the simplification of the project.

1. *Desk:* The desk used in the project was a black desk. However, any colour of desk with high saturation could still work, but there has to be enough contrast between the paper document and the desk.

2. Paper documents: It was assumed that all paper documents on the desk are of the same size (A4). The paper documents used are single front pages which consists of the title of published journals or articles and textbooks. Also the digital versions of the physical documents consist of an image database of the physical document.

### 4.2.2 Recognition and Localization

The first step to recognizing paper documents is finding the four sides of the document. As mentioned earlier, contrast between the paper document and the desk is the key to finding the corners of the paper document. The input video from the camera has to first of all be pre-processed for better recognition of the paper documents. The paper document recognition stage is also divided into several steps.

1. **Converting to gray-scale:** The video frames from the camera are first of all converted into gray-scale. A gray-scale image is an image in which each pixel in the image carries only intensity information. Converting the video frames to gray-scale is very important because most of the vision-based recognition and identification algorithms are based on pixel intensities of the video frame. The OpenCV *cvtColor* function was used for the conversion to gray-scale.

2. **Smoothing:** Smoothing of the gray-scale image is done to reduce noise or camera artifacts, reduce the resolution of the image frame and extract interesting visual features. Smoothing is predominantly done by applying a filter function to the video frames. Filtering analyzes the image by looking at the gray-level variations that are present and then decomposes the image into its frequency content from the lowest to the highest frequency. Low frequencies correspond to areas where the image intensities vary slowly while high frequencies are generated by rapid changes in intensities. OpenCV offers different filtering functions, however after some tests the Gaussian filter was found to be the most efficient filter for the job.

3. **Thresholding:** Once the video frames are converted into gray scale, the next step is *thresholding* the gray-scale video frames. Thresholding helps to separate out regions containing the paper documents which

are the regions we want to analyze. The separation is based on the variation of pixel intensity between the paper document and the desk surface. To differentiate the pixels of the paper documents from those of the desk, a comparison of each pixel intensity value with respect to a predetermined threshold value is done. The OpenCV *threshold* function was used for thresholding.

4. **Edge Detection** Edge detection follows immediately after thresholding. An edge in an image in computer vision is a sharp variation of the pixel intensity in an image. Edge detection is the core to recognizing paper documents on the desk. The Opencv *findcontours* function was used for edge detection. Detecting the edges of the video frame distinguishes the corners of the paper documents on the desk. Using the edge points (which are the corners of the paper document) detected in the video frames, an approximate rectangle can be drawn, localized and extracted from the video frames.

5. **Localization and Extraction:** Using the edge points detected, an approximate rectangle can be drawn, and the vertex and centre point of the drawn rectangle are then saved as a recognized document object in the system. The document recognition stage is finalized by the extraction of the recognized document. figure 32 shows the resulting image after localization.

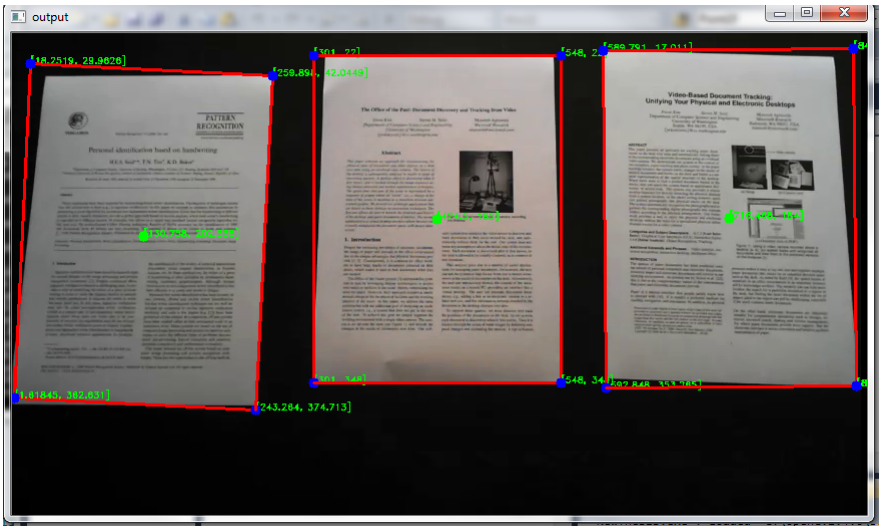


Figure 32: *Resulting image after edge detection and localization*

Extraction is a necessary and important step especially as it has significant use for document identification. The images of the paper document captured by the camera are not typically upright for high yield ORC processing or any similar processing which require to get the best possible features from the image to compare with a predefined template. The image needs to be transformed into a suitable upright orientation. OpenCV offers two functions for doing this namely *getPerspectiveTransform* and *warpPerspective*. The underlying principle behind this function is the linear transformation from one image space to another using a 3X3 transform matrix. The process can be better demonstrated in Figure

### 4.2.3 Identification

Identification of the recognized paper documents is of great importance for tracking paper documents. The digital documents recognized from the recognition stage need to be identified as a unique document. Identification of document is somewhat challenging because the recognized document from the video frames captured by the camera are not of good resolution. To uniquely identify the extracted document images, we need a metric that is unique to each document. Because it is highly unlikely for two distinct documents to have exactly the same structure, this means the variation of pixel intensity across each extracted paper document is unique. Therefore image histograms are used as a metric for the unique identification of the extracted document from the video frames. Image histogram is a graphical representation of the pixel intensity distribution of an image [52]. It quantifies the number of pixels for each intensity value considered. In the project, the Speed-Up Robust Feature(SURF) is used to accomplish the identification of the extracted documents.

SURF computes descriptive local features of an image based on histograms of edge orientation in a window around each point in the image. Three important characteristics of SURF make it the best method for the identification process.

1. *Invariance to 2D scale, translation and rotation:* When trying to match features across different images, scale changes is always a problem encountered. Scale change problems can be observed when processing different images taken at a different distance from the objects of interests. The paper documents present in the video frames from the camera are at different distance from the images of the paper document stored

in memory. Since the identification process is done by matching the local features of the paper document present in the video frames from the over head camera with the features of the paper document images present in the database, the SURF features which are scale-invariant are well suited for the purpose of identification.

2. *Distinctiveness:* SURF has a high-dimensional (128-D) descriptor which enables it to accurately differentiate between large numbers of features.

3. *Robust matching:* Detection and matching is robust with respect to partial occlusion and differences in contrast and illumination.

   SURF uses a basic second-order Hessian matrix approximation with box filters for feature point detection. The SURF algorithm constructs a scale image pyramid, by dividing the scale space into octaves with 4 scale level per octave as shown in Figure 33. Each octave represents a series of filter response maps obtained by convolving the same input image with a filter of increasing size. The minimum scale difference between two subsequent scales depends on the length of the positive or negative lobes of the partial second order derivative in the direction of derivation. Do non-maximum suppression in a 3 x 3 x 3 neighbourhood to get the steady feature points and the scale of values. To accommo-

   date for invariance to image rotation, the Haar wavelet responses are calculated in $x$ and $y$ direction within a circular neighborhood of radius $6s$ around the feature point, $s$ is the scale at which the feature point was detected. The Haar wavelet responses are represented as vectors. Then sum all the vector of $x$ and $y$ direction of the Haar wavelet responses within a sliding orientation window covering an angle of size $\pi/3$ around the feature point. The two summed response yield a new vector. And the longest vector is the dominant orientation of the feature point. For extraction of thedescriptor, construct a square region

   with a size of $20s$ and split the interest region up into a 4 x 4 square sub-regions with 5 x 5 regularly spaced sample points inside. As shown in Figure 33, compute the Haar wavelet response x-direction $d_x$ and the Haar wavelet response y-direction $d_y$. Weigh the response with a Gaussian kernel centered at the interest point. Sum the response over each sub-region for $d_x$ and $d_y$ separately. In order to bring in information about the polarity of the intensity changes, extract the sum of absolute value of the responses. Therefore, each sub-region is formed

a4-dimensional vector,

$$Vec = \left(\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|\right)$$

Finally, normalize the vector into unit length for invariance to contrast.

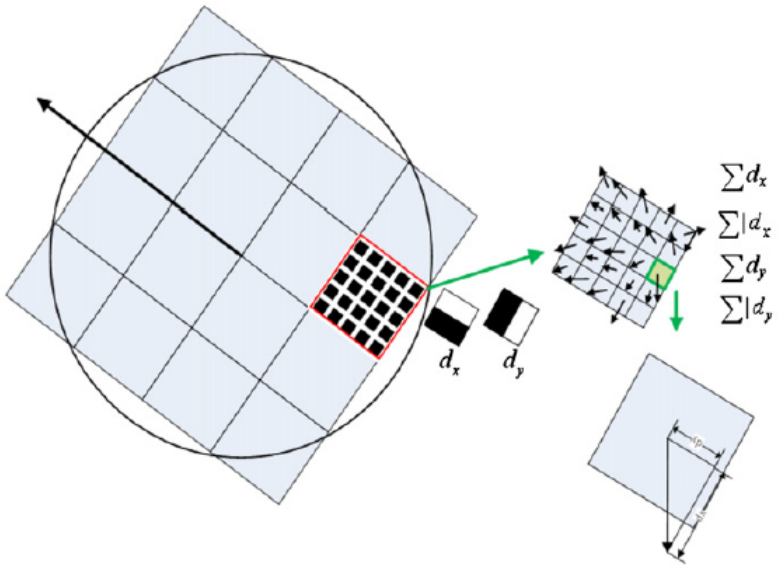


Figure 33: *Feature descriptor of SURF*

A flow diagram of the above process is illustrated in Figure 34.

### 4.2.4 Pile of Paper Documents

The recognition and identification of paper documents in a pile is the central goal of this research. In an ideal case of a pile, all the paper documents are placed on top of each other with minimum overlapping borders. The main challenge for paper document in a pile arises in situations where paper documents are placed haphazardly on the desk (see Figure 35 which makes it impossible to distinguish between piles.

As an attempt to solve this issue, some assumptions were made to model a paper document to belong to a pile. The euclidean distance between the centre points of the paper documents were used as a metric to categorize paper documents into different piles in the digital model. It was assumed that for two paper documents to belong to the same pile, the euclidean distance

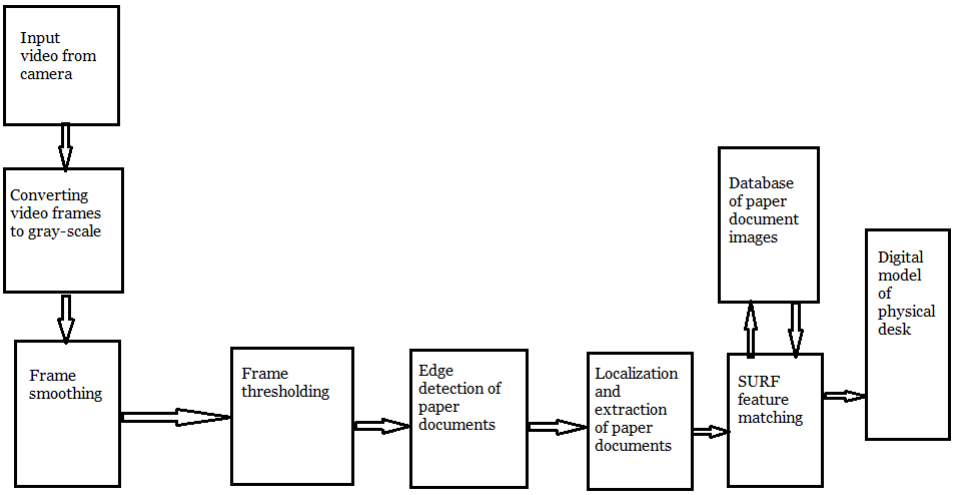


Figure 34: *Flow diagram of over all process*

between their centre points must be less than the width of a paper document (we assume that all paper documents have the same size). Figure 36 illustrates this.

Each pile is also modeled as having a unique identifier called pileID. PileIDs are simply positive integer numbers. In the spacial case illustrated in Figure 37, doc A will be added to pile 1 if and only if $d1 < w/2$ and $d1 < d2$. Likewise, doc A will be added to pile 2 if and only if $d2 < w/2$ and $d2 < d1$. In the rare cases where $d1 < w/2$,$d2 < w/2$ and $d1 < d2$, doc A will be added to pile 1 if the pileID of pile 1 is less than the PileID of pile 2 and likewise, doc A will be added to pile 2 if the pileID of pile 2 is less than the pileID of pile 1.

Three common scenarios of paper document piles were handled.

1. The first scenario occurs when the piles of paper documents are spatially distinct from each other as in the case of Figure 38. All the documents in this scenario are modeled to belong to either pile 1, pile 2, or pile 2.

2. The second scenario occurs when a document is placed to overlap between two distinct piles as shown in Figure 39. Such a scenario is modeled as three separate piles if non of the conditions explained in Figure 37 are satisfied.

3. The third scenario occurs when the paper documents are placed hap-

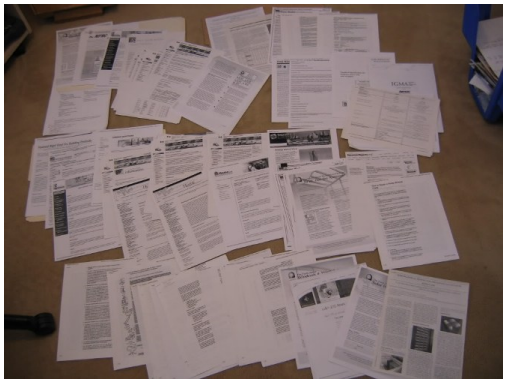

Figure 35: *Haphazard paper documents arrangement*

hazardly on the desk, with no clear distinction between piles as shown in Figure 40. In such a scenario, the conditions for categorizing paper documents in a pile will be applied, but the digital model would not accurately represent the physical localization of the documents. As shown in Figure 40 this scenario will be modeled based on the conditions as belonging to 4 piles which may not be the perfect representation of the physical world.

To handle to case of documents in piles, successive input frames are analyzed to determine movements of paper documents on the desk. The frame difference is computed between consecutive input frames and if there is a large difference, it is assumed that a document is moving on the desk. To determine a valid movement if paper document, the SURF features in the video frame before the motion of the document starts and the video frame immediately after the motion stops are matched with the SURF features from the image database stored in memory. The pairs of matching features that have similar motion are clustered. If the largest cluster with a non-zero motion contains sufficiently many matches, it is considered a valid motion of paper document, and the paper document is identified and tracked.

### 4.2.5 Results

In this section we discuss the results and present a performance analysis on document recognition. The system was tested with 40 printed paper documents. The printed documents consisted of the first pages pages of articles, cover pages or report documents and cover pages of text books. The experiment was done in real time. The desk was initially empty and paper documents were slowly added to the desk forming three paper document

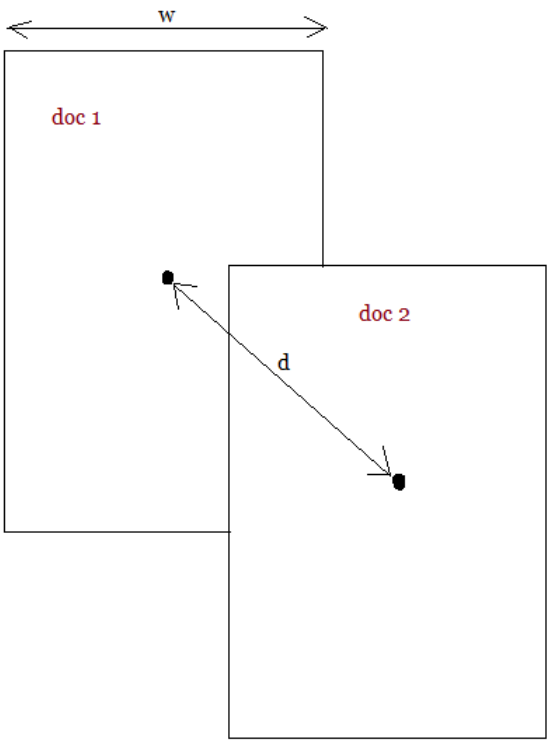


Figure 36: *Euclidean distance as a metric to categorize documents in different piles.*
*doc 1 and doc 2 belong to the same pile if* $d < w/2$

piles. The image database containing 40 images (in JPEG format) of the paper documents were used in the experiments. The images in the database were approximately 500 x 647 pixels (width x height). Due to the relatively slow running SURF algorithm, it took about 6 to 7 minutes for the 40 paper documents to be placed on the desk for recognition and identification. This experiment was done in about 8 times. The results can be shown in Table 41 and figure 42.

From the results, 6 out of the 8 experiments, all 40 documents were recognized. However, there were a lot of falsely recognized paper documents. In all the experiments, all the documents consisting of more graphical and drawings were correctly recognized. Documents which contains pure text were falsely recognized most of the time.

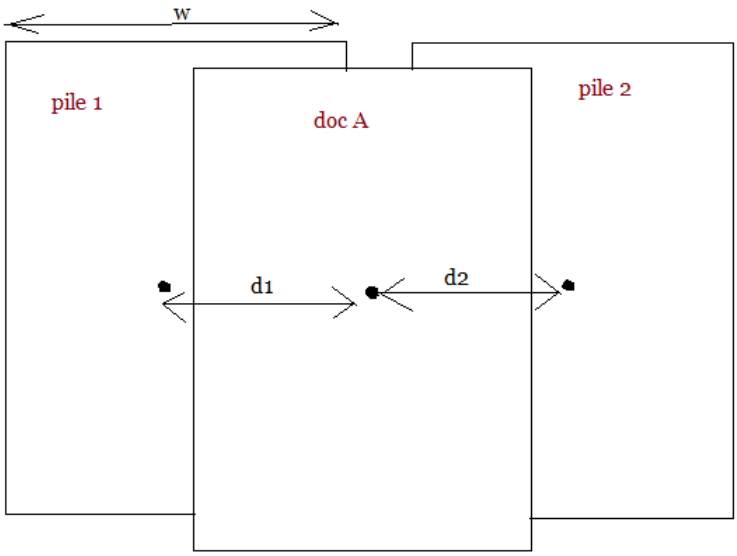


Figure 37: *Modeling of paper document piles in spacial cases*

### 4.2.6 Technical Issues

There were a few difficulties encountered in the course of implementation. The first issue faced was varying light intensity. The experiments on the system were carried out indoor, with a fair amount of indoor lighting. However, a slight variation of light intensity highly affected the threshold value used in the thresholding stage, which in term affected the paper document recognition phase. A sudden increase in light intensity increases the overall average pixel intensity of the video frames from the camera. A possible way to solve this issue is to design the system such that the threshold value automatically adjusts to the varying light intensity. Also the light intensity affects the SURF algorithm used for document recognition. This issue was a constant recurrence during testing where there was an alarming rate of false positive recognition. With a very high light intensity, for example like a beam of sunlight reflecting on the desk containing the documents, only 5 out of the 40 documents were correctly matched with the correct images in the database.

The second issue encountered was the slow running of the SURF recognition algorithm. This is a very disturbing issue since the system needs to

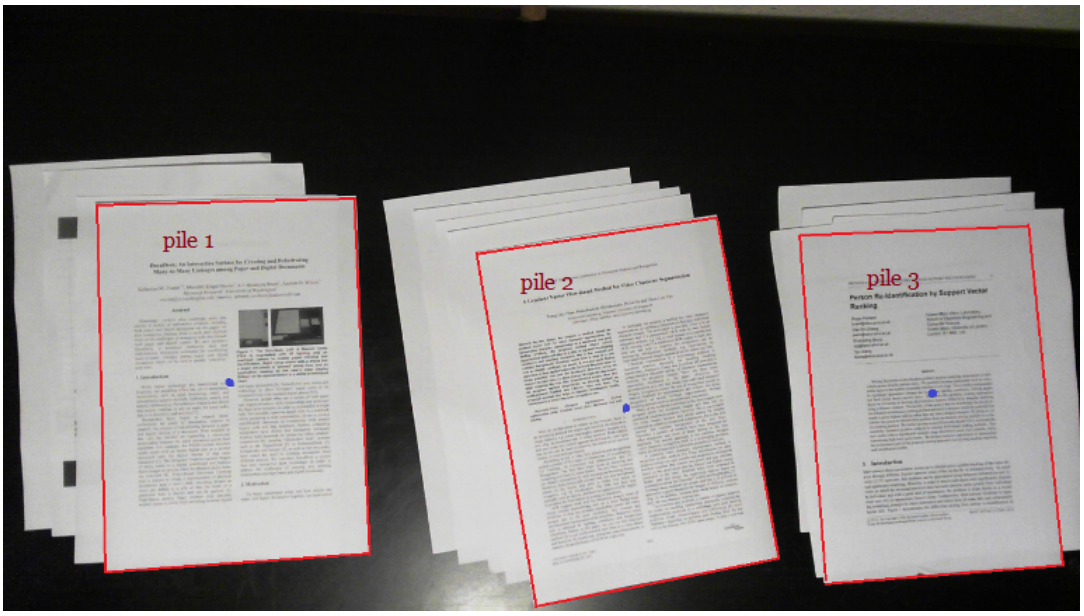


Figure 38: *Paper documents arranged into three distinct piles*

operate in real time and changes in the physical world need to be immediately reflected in the digital world. As an attempt to solve this issue, the size of the input frame was reduced and the threshold for a point to be categorized as a feature descriptive point was increased so to limit the size of the SURF descriptor used for identification. Despite all these, the algorithm still runs relatively slow for real time recognition and tracking. Due to the slow running algorithm, the paper documents were placed very slowly on the desk giving at least a 6 seconds interval between the placement of each paper document.

It was also realized that for a better recognition and identification and tracking, the paper documents have to be moved close to the overhead camera for a better image resolution before being moved slowly on the desk. This is an expected behaviour because more SURF features will be detected the documents are close to the camera and hence there will be more features to match against the database of images. When the paper documents were moved closed to the camera before being placed on the desk, 20 out of the 40 paper documents were perfectly matched to the images in the database.

The third issue encountered was the problem of camera auto focus. The camera used for the implementation of the thesis had a built in auto focus,

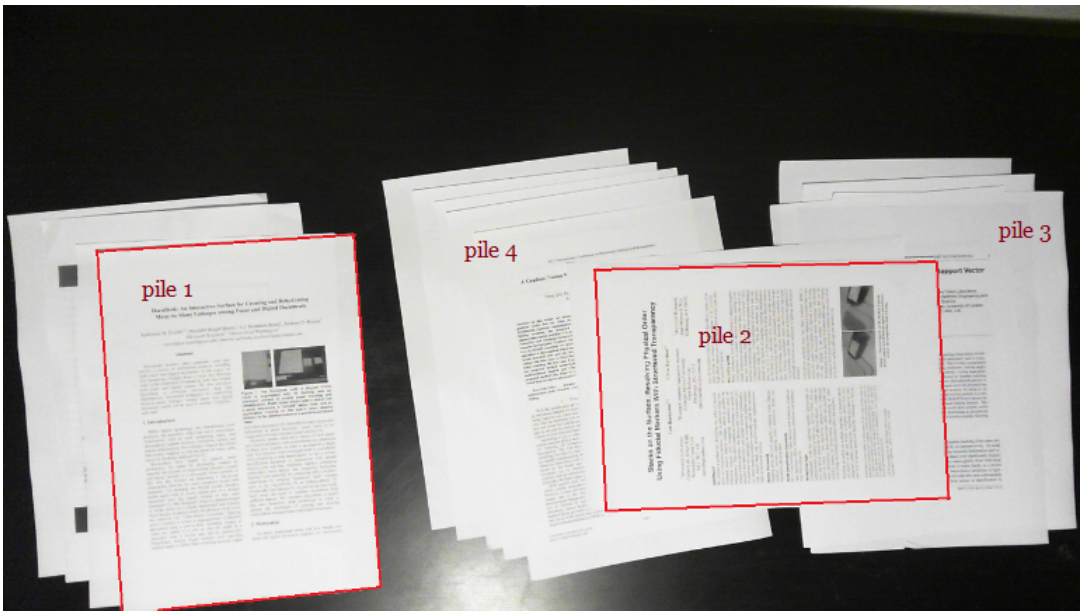


Figure 39: *Special case: Paper document overlapping between two distinct piles*

which was really a big issue because most of the time, especially when paper document is added, removed or moved from the desk, the input video frames get blurred. This seriously affects the entire recognition and identification process and leads to an alarming rate of false recognition and identification.

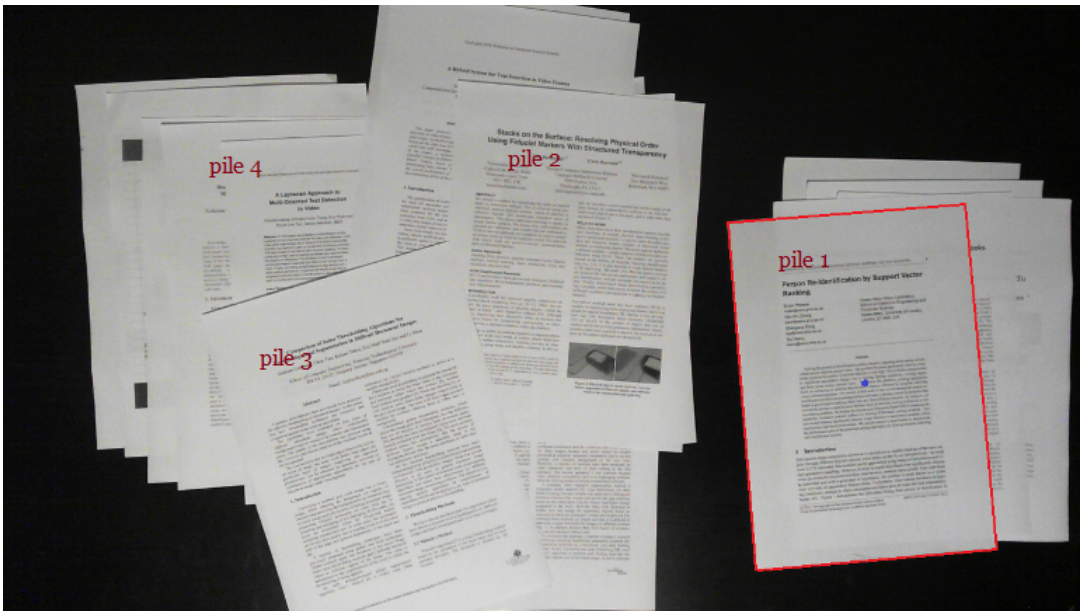


Figure 40: *Special case: Indistinguishable paper document piles*

# 5 Conclusion and Future Work

This thesis presents a digital desk system for seamless integration of physical paper document and digital document by the recognition, localization, identification and tracking of paper document on the office desk without requiring a new physical infrastructure besides a digital video camera.

The approach which was used by the system to identify paper documents from video frames was calculating Scale-invariant descriptive local features in the image frames from the camera, matching them with an image database of paper documents.

So far, the system can only accurately recognize and identify paper documents consisting of a more graphical and distinct layout. The system could be improved to better recognize paper documents consisting of purely text.

A useful improvement would be to add query mechanism, so that the user could query the system to search for documents on the desk. This system would give the exact location of the document on the desk.

Another useful addition to the system would be detecting changes to the paper document surface when users make written annotation. The written annotation may be automatically captured by the system and incorporated

| Experiment number | Correctly recognized | Falsely recognized | Not recognized | Precision | Recall |
|---|---|---|---|---|---|
| 1 | 15 | 35 | 0 | 3/8 | 3/8 |
| 2 | 10 | 40 | 0 | 1/4 | 1/4 |
| 3 | 15 | 35 | 0 | 3/8 | 1/4 |
| 4 | 14 | 35 | 1 | 14/39 | 7/10 |
| 5 | 12 | 38 | 0 | 3/10 | 3/10 |
| 6 | 12 | 38 | 0 | 3/10 | 3/10 |
| 7 | 14 | 34 | 2 | 7/19 | 7/10 |
| 8 | 15 | 35 | 0 | 3/8 | 3/8 |

Figure 41: *Paper document recognition experimental results*

into the digital version of the paper document stored in memory.

In conclusion, the concept of the digital desk which is the integration of the digital and physical workspace to allow users to benefit from the unlimited advantages of the digital world while using their native physical manipulative skills to manipulate digital objects. The perfect digital desk will guarantee an increase in productivity in the workplace and a better personal information management, since finding a document on the desk will be just a click (or gesture, or speech) away.
This work adds to the body of work towards this future perfect digital desk with the following contribution:

- The study of OCR engines to verity if it is a good approach for paper document recognition of using a video input stream of a camera in real time.
- Implementation of document recognition and tracking system using the state-of-the-art computer vision algorithm in feature detection and matching (SURF) as a solution for real time recognition and tracking.
- The implementation issues to be considered when implementing such a system that recognizes, identifies and tracks paper documents using a digital video camera in real time.

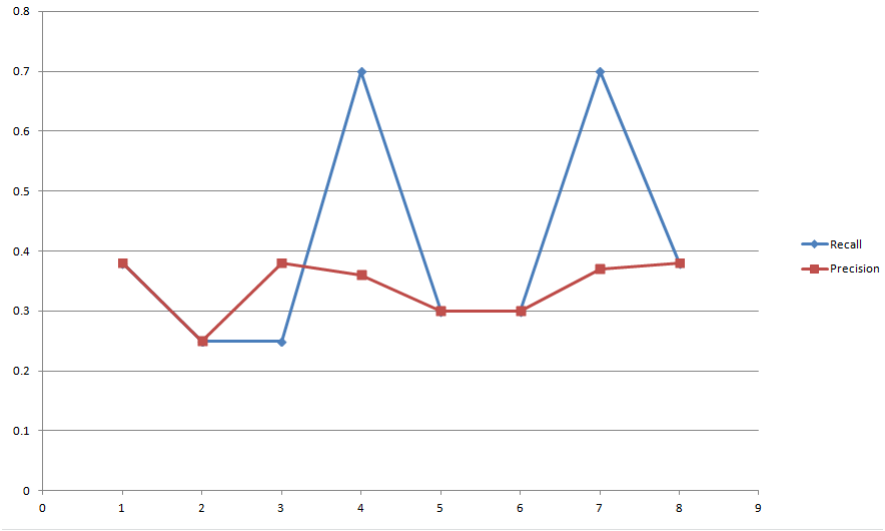


Figure 42: *Paper document recognition experimental results: precision and recall*

# References

[1] Marios Anthimopoulos, Basilios Gatos, and Ioannis Pratikakis. A hybrid system for text detection in video frames. In *Document Analysis Systems, 2008. DAS'08. The Eighth IAPR International Workshop on*, pages 286–292. IEEE, 2008.

[2] Hiroki Arakawa. On-line recognition of handwritten charactersalphanumerics, hiragana, katakana, kanji. *Pattern Recognition*, 16(1):9–21, 1983.

[3] Nafiz Arica and Fatos T Yarman-Vural. One-dimensional representation of two-dimensional information for hmm based handwriting recognition. *Pattern Recognition Letters*, 21(6):583–592, 2000.

[4] C Bradford Barber, David P Dobkin, and Hannu Huhdanpaa. The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software (TOMS)*, 22(4):469–483, 1996.

[5] Issam Bazzi, Richard Schwartz, and John Makhoul. An omnifont open-vocabulary ocr system for english and arabic. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(6):495–504, 1999.

[6] Ofer Bergman, Richard Boardman, Jacek Gwizdka, and William Jones. Personal information management. In *Chi'04 extended abstracts on human factors in computing systems*, pages 1598–1599. ACM, 2004.

[7] Dennis B Beringer and James G Peterson. Underlying behavioral parameters of the operation of touch-input devices: Biases, models, and feedback. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 27(4):445–458, 1985.

[8] Xiaojun Bi, Tovi Grossman, Justin Matejka, and George Fitzmaurice. Magic desk: bringing multi-touch surfaces into desktop work. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 2511–2520. ACM, 2011.

[9] Barry Blesser. Multistage digital filtering utilizing several criteria, February 1983. US Patent 4,375,081.

[10] Jean-Luc Bloechle. *Physical and logical structure recognition of PDF documents*. PhD thesis, Doctoral thesis, 2010.

[11] Mindy Bokser. Omnidocument technologies. *Proceedings of the IEEE*, 80(7):1066–1078, 1992.

[12] Otto Bretscher. *Linear algebra with applications*. Prentice Hall Englewood Cliffs, NJ, 1997.

[13] Thomas M Breuel. The ocropus open source ocr system. *DRR*, 6815:68150, 2008.

[14] Heather Brown and Peter Robinson. Integrating paper and digital documents. In *Digital Media: The Future*, pages 128–143. Springer, 2000.

[15] MK Brown and S Ganapathy. Preprocessing techniques for cursive script word recognition. 1983.

[16] CF Chen and CH Hsiao. Haar wavelet method for solving lumped and distributed-parameter systems. In *Control Theory and Applications, IEE Proceedings-*, volume 144, pages 87–94. IET, 1997.

[17] Ming Chen and Xiaoqing Ding. A robust skew detection algorithm for grayscale document image. In *Document Analysis and Recognition, 1999. ICDAR'99. Proceedings of the Fifth International Conference on*, pages 617–620. IEEE, 1999.

[18] K Chidananda Gowda and G Krishna. Agglomerative clustering using the concept of mutual nearest neighbourhood. *Pattern Recognition*, 10(2):105–112, 1978.

[19] Anthony Creed, Ian Dennis, and Stephen Newstead. Proof-reading on vdus. *Behaviour & Information Technology*, 6(1):3–13, 1987.

[20] Andrew Dillon. Reading from paper versus screens: A critical review of the empirical literature. *Ergonomics*, 35(10):1297–1326, 1992.

[21] Andrew Dillon, John Richardson, and Cliff McKnight. Navigation in hypertext: a critical review of the concept. *Navigation*, 1990.

[22] Viet Cuong Dinh, Seong Soo Chun, Seungwook Cha, Hanjin Ryu, and Sanghoon Sull. An efficient method for text detection in video based on stroke width similarity. In *Computer Vision–ACCV 2007*, pages 200–209. Springer, 2007.

[23] David Doermann, Jian Liang, and Huiping Li. Progress in camera-based document image analysis. In *Document Analysis and Recognition, 2003. Proceedings. Seventh International Conference on*, pages 606–616. IEEE, 2003.

[24] Line Eikvil. Optical character recognition. *citeseer. ist. psu. edu/142042. html*, 1993.

[25] Ame Elliott and Marti A Hearst. How large should a digital desk be?: qualitative results of a comparative study. In *CHI'00 extended abstracts on Human factors in computing systems*, pages 165–166. ACM, 2000.

[26] Katherine M Everitt, Meredith Ringel Morris, AJ Bernheim Brush, and Andrew D Wilson. Docudesk: An interactive surface for creating and rehydrating many-to-many linkages among paper and digital documents. In *Horizontal Interactive Human Computer Systems, 2008. TABLETOP 2008. 3rd IEEE International Workshop on*, pages 25–28. IEEE, 2008.

[27] George W Fitzmaurice, Hiroshi Ishii, and William AS Buxton. Bricks: laying the foundations for graspable user interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 442–449. ACM Press/Addison-Wesley Publishing Co., 1995.

[28] Hiromichi Fujisawa, Yasuaki Nakano, and Kiyomichi Kurino. Segmentation methods for character recognition: from segmentation to document structure analysis. *Proceedings of the IEEE*, 80(7):1079–1092, 1992.

[29] John D Gould, Lizette Alfaro, Vincent Barnes, Rich Finn, Nancy Grischkowsky, and Angela Minuto. Reading is slower from crt displays than from paper: attempts to isolate a single-variable explanation. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 29(3):269–299, 1987.

[30] John D Gould and Nancy Grischkowsky. Doing the same work with hard copy and with cathode-ray tube (crt) computer terminals. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 26(3):323–337, 1984.

[31] Chen Yan Graham Leedham, Kalyan Takru, Joie Hadi Nata Tan, and Li Mian. Comparison of some thresholding algorithms for text/background segmentation in difficult document images. In *Proceedings of the seventh international conference on document analysis and recognition*, volume 2, pages 859–864. Citeseer, 2003.

[32] Gabriel F Groner. Real-time recognition of handprinted text. In *Proceedings of the November 7-10, 1966, fall joint computer conference*, pages 591–601. ACM, 1966.

[33] Wacef Guerfali and Réjean Plamondon. Normalizing and restoring online handwriting. *Pattern Recognition*, 26(3):419–431, 1993.

[34] Didier Guillevic and Ching Y Suen. Cursive script recognition: A sentence level recognition scheme. In *Proc. 4th IWFHR*, pages 216–223. Citeseer, 1994.

[35] François Guimbretière. Paper augmented digital documents. In *Proceedings of the 16th annual ACM symposium on User interface software and technology*, pages 51–60. ACM, 2003.

[36] Christina Haas. *Writing technology: Studies on the materiality of literacy*. Psychology Press, 1996.

[37] Michael Haller, Jakob Leitner, Thomas Seifried, James R Wallace, Stacey D Scott, Christoph Richter, Peter Brandl, Adam Gokcezade, and Seth Hunter. The nice discussion room: Integrating paper and digital media to support co-located group meetings. In *Proceedings*

*of the SIGCHI Conference on Human Factors in Computing Systems*, pages 609–618. ACM, 2010.

[38] Jefferson Y Han. Low-cost multi-touch sensing through frustrated total internal reflection. In *Proceedings of the 18th annual ACM symposium on User interface software and technology*, pages 115–118. ACM, 2005.

[39] Akihide Hashizume, Pen-Shu Yeh, and Azriel Rosenfeld. A method of detecting the orientation of aligned components. *Pattern Recognition Letters*, 4(2):125–132, 1986.

[40] David L Hecht. Embedded data glyph technology for hardcopy digital documents. In *Proc. Society of Photo-Optical Instrumentation Engineers Symp. on Electronic Imaging, Science and Technology, San Jose, Calif*, volume 2171, pages 341–352, 1994.

[41] Weihua Huang, Palaiahnakote Shivakumara, and Chew Lim Tan. Detecting moving text in video using temporal information. In *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, pages 1–4. IEEE, 2008.

[42] James R Janesick. *Scientific charge-coupled devices*, volume 117. SPIE press Bellingham, WA, 2001.

[43] Zhong Ji, Jian Wang, and Yu-Ting Su. Text detection in video frames using hybrid features. In *Machine Learning and Cybernetics, 2009 International Conference on*, volume 1, pages 318–322. IEEE, 2009.

[44] Cheolkon Jung, Qifeng Liu, and Joongkyu Kim. A new approach for text segmentation using a stroke filter. *Signal Processing*, 88(7):1907–1916, 2008.

[45] Jiwon Kim, Steven M Seitz, and Maneesh Agrawala. The office of the past: Document discovery and tracking from video. In *Computer Vision and Pattern Recognition Workshop, 2004. CVPRW'04. Conference on*, pages 157–157. IEEE, 2004.

[46] Jiwon Kim, Steven M Seitz, and Maneesh Agrawala. Video-based document tracking: unifying your physical and electronic desktops. In *Proceedings of the 17th annual ACM symposium on User interface software and technology*, pages 99–107. ACM, 2004.

[47] KC Knowlton. Computer displays optically superimposed on input devices. *Bell System Technical Journal*, 56(3):367–383, 1977.

[48] András Kornai, KM Mohiuddin, and Scott D Connell. Recognition of cursive writing on personal checks. In *Proc 5th International Workshop on Frontiers in Handwriting Recognition, Essex, UK*, pages 373–378, 1996.

[49] Myron W Krueger. *Artificial reality II*, volume 10. Addison-Wesley Reading (Ma), 1991.

[50] Myron W Krueger, Thomas Gionfriddo, and Katrin Hinrichsen. Videoplacean artificial reality. In *ACM SIGCHI Bulletin*, volume 16, pages 35–40. ACM, 1985.

[51] Karen Kukich. Techniques for automatically correcting words in text. *ACM Computing Surveys (CSUR)*, 24(4):377–439, 1992.

[52] Robert Laganière. *OpenCV 2 computer vision application programming cookbook*. Packt Publishing, 2011.

[53] Rainer Lienhart and Axel Wernicke. Localizing and segmenting text in images and videos. *Circuits and Systems for Video Technology, IEEE Transactions on*, 12(4):256–268, 2002.

[54] Jaeguyn Lim, Jonghyun Park, and Gérard G Medioni. Text segmentation in color images using tensor voting. *Image and Vision Computing*, 25(5):671–685, 2007.

[55] David G Lowe. Object recognition from local scale-invariant features. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, volume 2, pages 1150–1157. Ieee, 1999.

[56] WW Loy and ID Landau. An on-line procedure for recognition of handprinted alphanumeric characters. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (4):422–427, 1982.

[57] Sriganesh Madhvanath, Gyeonghwan Kim, and Venu Govindaraju. Chaincode contour processing for handwritten word recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(9):928–932, 1999.

[58] Thomas W Malone. How do people organize their desks?: Implications for the design of office information systems. *ACM Transactions on Information Systems (TOIS)*, 1(1):99–112, 1983.

[59] Ian C McIlwaine. The universal decimal classification: Some factors concerning its origins, development, and influence. *JASIS*, 48(4):331–339, 1997.

[60] Darnell Janssen Moore, Irfan A Essa, and Monson H Hayes. Object spaces: Context management for human activity recognition. 1998.

[61] Paul Muter, Susane A Latrémouille, William C Treurniet, and Paul Beam. Extended reading of continuous text on television screens. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 24(5):501–508, 1982.

[62] Yinan Na and Di Wen. An effective video text tracking algorithm based on sift feature and geometric constraint. In *Advances in Multimedia Information Processing-PCM 2010*, pages 392–403. Springer, 2010.

[63] Randal C Nelson and Isaac A Green. Tracking objects using recognition. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, volume 2, pages 1025–1030. IEEE, 2002.

[64] William Newman and Pierre Wellner. A desk supporting computer-based interaction with paper documents. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '92, pages 587–592. ACM, 1992.

[65] Konstantinos Ntirogiannis, Basilios Gatos, and Ioannis Pratikakis. Binarization of textual content in video frames. In *Document Analysis and Recognition (ICDAR), 2011 International Conference on*, pages 673–677. IEEE, 2011.

[66] Sophie Oetjen and Martina Ziefle. A visual ergonomic evaluation of different screen types and screen technologies with respect to discrimination performance. *Applied ergonomics*, 40(1):69–81, 2009.

[67] Kenton O'Hara and Abigail Sellen. A comparison of reading paper and on-line documents. In *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems*, pages 335–342. ACM, 1997.

[68] Jennifer Pearson, George Buchanan, Harold Thimbleby, and Matt Jones. The digital reading desk: A lightweight approach to digital note-taking. *Interacting with Computers*, 24(5):327–338, 2012.

[69] Xujun Peng, Huaigu Cao, Rohit Prasad, and Premkumar Natarajan. Text extraction from video using conditional random fields. In *Document Analysis and Recognition (ICDAR), 2011 International Conference on*, pages 1029–1033. IEEE, 2011.

[70] Trung Quy Phan, Palaiahnakote Shivakumara, Bolan Su, and Chew Lim Tan. A gradient vector flow-based method for video character segmentation. In *Document Analysis and Recognition (ICDAR), 2011 International Conference on*, pages 1024–1028. IEEE, 2011.

[71] Jennie P Psihogios, Carolyn M Sommerich, Gary A Mirka, and Samuel D Moon. A field evaluation of monitor placement effects in vdt users. *Applied Ergonomics*, 32(4):313–325, 2001.

[72] Jun Rekimoto and Yuji Ayatsuka. Cybercode: Designing augmented reality environments with visual tags. In *Proceedings of DARE 2000 on Designing Augmented Reality Environments*, DARE '00, pages 1–10, New York, NY, USA, 2000. ACM.

[73] John Richardson, A Dillon, C McKnight, and M Saadat-Samardi. *The Manipulation of Screen-presented Text: Experimental Investigation of an Interface Incorporating a" movement Grammar"*. Human Sciences and Advanced Technology, 1988.

[74] Daniela Rus and Peter De Santis. The self-organizing desk. In *In Proceedings of the International Joint Conference on Artificial Intelligence*. Citeseer, 1997.

[75] Nabin Sharma, Umapada Pal, and Michael Blumenstein. Recent advances in video based document processing: a review. In *Document Analysis Systems (DAS), 2012 10th IAPR International Workshop on*, pages 63–68. IEEE, 2012.

[76] Palaiahnakote Shivakumara, Souvik Bhowmick, Bolan Su, Chew Lim Tan, and Umapada Pal. A new gradient based character segmentation method for video text recognition. In *Document Analysis and Recognition (ICDAR), 2011 International Conference on*, pages 126–130. IEEE, 2011.

[77] Palaiahnakote Shivakumara, Anjan Dutta, Trung Quy Phan, Chew Lim Tan, and Umapada Pal. A novel mutual nearest neighbor based symmetry for text frame classification in video. *Pattern Recognition*, 44(8):1671–1683, 2011.

[78] Palaiahnakote Shivakumara, Weihua Huang, and Chew Lim Tan. An efficient edge based technique for text detection in video frames. In *Document Analysis Systems, 2008. DAS'08. The Eighth IAPR International Workshop on*, pages 307–314. IEEE, 2008.

[79] Palaiahnakote Shivakumara, Trung Quy Phan, and Chew Lim Tan. A robust wavelet transform based technique for video text detection. In *Document Analysis and Recognition, 2009. ICDAR'09. 10th International Conference on*, pages 1285–1289. IEEE, 2009.

[80] Palaiahnakote Shivakumara, Trung Quy Phan, and Chew Lim Tan. A laplacian approach to multi-oriented text detection in video. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(2):412–419, 2011.

[81] Palaiahnakote Shivakumara and Chew Lim Tan. Novel edge features for text frame classification in video. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 3191–3194. IEEE, 2010.

[82] Beat Signer and Moira C Norrie. Interactive paper: past, present and future. In *proceedings of the 1st International Workshop on Paper Computing (PaperComp 2010)*, 2010.

[83] Itiro Siio, Jim Rowan, and Elizabeth Mynatt. Finding objects in strata drawer. In *CHI'03 extended abstracts on Human factors in computing systems*, pages 982–983. ACM, 2003.

[84] Ray Smith. An overview of the tesseract ocr engine. In *Document Analysis and Recognition, 2007. ICDAR 2007. Ninth International Conference on*, volume 2, pages 629–633. IEEE, 2007.

[85] Robert F Sproull. Using program transformations to derive line-drawing algorithms. *ACM Transactions on Graphics (TOG)*, 1(4):259–273, 1982.

[86] Jürgen Steimle, Mohammadreza Khalilbeigi, and Max Mühlhäuser. Hybrid groups of printed and digital documents on tabletops: a study. In *CHI'10 Extended Abstracts on Human Factors in Computing Systems*, pages 3271–3276. ACM, 2010.

[87] John C Tang and Scott L Minneman. Videodraw: a video interface for collaborative drawing. *ACM Transactions on Information Systems (TOIS)*, 9(2):170–184, 1991.

[88] Bruce Tognazzini. The starfire video prototype project: a case history. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 99–105. ACM, 1994.

[89] Sandra Truellemans. Personal cross-media information management. Master's thesis, Vrije Universiteit Brussels, 2013.

[90] Endel Tulving and Donald M Thomson. Encoding specificity and retrieval processes in episodic memory. *Psychological review*, 80(5):352, 1973.

[91] Seiichi Uchida, Yuki Shigeyoshi, Yasuhiro Kunishige, and Feng Yaokai. A keypoint-based approach toward scenery character detection. In *Document Analysis and Recognition (ICDAR), 2011 International Conference on*, pages 819–823. IEEE, 2011.

[92] JV Vincent and F MacDougall. The mandala system, chi'90 interactive experience, april 1-5 1990. *Seattle, Washington*, 1990.

[93] Roy Want, Kenneth P Fishkin, Anuj Gujar, and Beverly L Harrison. Bridging physical and virtual worlds with electronic tags. In *CHI*, volume 99, pages 370–377, 1999.

[94] Roy Want, Kenneth P. Fishkin, Anuj Gujar, and Beverly L. Harrison. Bridging physical and virtual worlds with electronic tags. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '99, pages 370–377, New York, NY, USA, 1999. ACM.

[95] Nadir Weibel, Adriana Ispas, Beat Signer, and Moira C Norrie. Paperproof: a paper-digital proof-editing system. In *CHI'08 extended abstracts on Human factors in computing systems*, pages 2349–2354. ACM, 2008.

[96] Mark Weiser. The computer for the 21st century. *Scientific american*, 265(3):94–104, 1991.

[97] Malte Weiss, Simon Voelker, and Jan Borchers. Benddesk: Seamless integration of horizontal and vertical multi-touch surfaces in desk environments. *Adjunct Proceedings ITS*, 9, 2009.

[98] Pierre Wellner. Interacting with paper on the digitaldesk. *Commun. ACM*, 36(7):87–96, July 1993.

[99] Raphael Wimmer, Fabian Hennecke, Florian Schulz, Sebastian Boring, Andreas Butz, and Heinrich Hußmann. Curve: revisiting the digital desk. In *Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries*, pages 561–570. ACM, 2010.

[100] Patricia Wright and Alan Lickorish. Proof-reading texts on screen and paper. *Behaviour & Information Technology*, 2(3):227–235, 1983.

[101] Chih-Sung Andy Wu, Susan J Robinson, and Ali Mazalek. Turning a page on the digital annotation of physical books. In *Proceedings of the 2nd international conference on Tangible and embedded interaction*, pages 109–116. ACM, 2008.

[102] Chenyang Xu and Jerry L Prince. Gradient vector flow: A new external force for snakes. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 66–71. IEEE, 1997.

[103] Qixiang Ye, Jianbin Jiao, Jun Huang, and Hua Yu. Text detection and restoration in natural scene images. *Journal of Visual Communication and Image Representation*, 18(6):504–513, 2007.

[104] Ron Yeh, Chunyuan Liao, Scott Klemmer, François Guimbretière, Brian Lee, Boyko Kakaradov, Jeannie Stamberger, and Andreas Paepcke. Butterflynet: a mobile capture and access system for field biology research. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 571–580. ACM, 2006.

[105] Bin Yu and Anil K Jain. A robust and fast skew detection algorithm for generic documents. *Pattern recognition*, 29(10):1599–1629, 1996.

[106] Xin Zhang and Fuchun Sun. Pulse coupled neural network edge-based algorithm for image text locating. *Tsinghua Science & Technology*, 16(1):22–30, 2011.

[107] Ming Zhao, Shutao Li, and James Kwok. Text detection in images using sparse representation with discriminative dictionaries. *Image and Vision Computing*, 28(12):1590–1599, 2010.